REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-01-0188

1. REPORT DATE *(DD-MM-YYYY)*
11–002

2. REPORT TYPE
Technical

3. DATES COVERED *(From - To)*

4. TITLE AND SUBTITLE

DEDUCTION WITH UNCERTAIN CONDITIONALS

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
0601152N

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

6. AUTHORS

P. G. Calabrese

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

SSC San Diego
San Diego, CA 92152-5001

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Office of Naval Research
800 North Quincy Street
Arlington, VA 22217–5000

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT

20090803043

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Uncertain conditional propositions or conditional events pose a special problem for deduction due to their three-valued nature—true, false or inapplicable. This three-valued-ness gives rise to several different kinds of deduction between conditionals depending upon the particular deductive relation being employed. There is a hierarchy of 13 deductive relations between conditionals built up from four elementary ones, which can be expressed in terms of Boolean relations on the components of the conditionals. These different, but interrelated deductive relations on conditionals in tum give rise to various deductively closed systems of conditionals. Theorems are proved relating thc deductive relations with each other and with their associated deductively closed sets (DCSs). The principal DCS generated by a single conditional using any of the deductive relations is completely characterized. Except for two of the deductive relations, deduction with a finite number of conditionals or with additional conditional information is also completely characterized as the union of associated principal DCSs. Examples of finite sets of deductively closed conditionals are exhibited including many, unlike Boolean algebra, which are finite and yet non-principal, that is, not generated by any one conditional. An introductory section carefully provides motivations and a complete algebraic characterization of the four chosen operations on conditional events. Results are applied to the famous penguin problem.

Published in *Information Sciences*, vol. 147, 2002, 143–191.

15. SUBJECT TERMS

Mission Area: Information Science

deduction	three-valued	probability	pre-order
uncertainty	logic	deductive relation	implication

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON / 19B. TELEPHONE NUMBER *(Include area code)*
U	U	U	UU	48	P. G. Calarese / (619) 553-3680

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

ELSEVIER

Information Sciences 147 (2002) 143–191

INFORMATION SCIENCES
AN INTERNATIONAL JOURNAL

www.elsevier.com/locate/ins

Deduction with uncertain conditionals ☆

Philip G. Calabrese

Joint and National Systems Division (Code 2737), Space and Naval Warfare Systems Center, San Diego, CA 92152-5001, USA

Received 13 June 2001; received in revised form 17 November 2001; accepted 1 April 2002

Communicated by Didier Dubois

Abstract

Uncertain conditional propositions or conditional events pose a special problem for deduction due to their three-valued nature – true, false or inapplicable. This three-valued-ness gives rise to several different kinds of deduction between conditionals depending upon the particular deductive relation being employed. There is a hierarchy of 13 deductive relations between conditionals built up from four elementary ones, which can be expressed in terms of Boolean relations on the components of the conditionals. These different, but interrelated deductive relations on conditionals in turn give rise to various deductively closed systems of conditionals. Theorems are proved relating the deductive relations with each other and with their associated deductively closed sets (DCSs). The principal DCS generated by a single conditional using any of the deductive relations is completely characterized. Except for two of the deductive relations, deduction with a finite number of conditionals or with additional conditional information is also completely characterized as the union of associated principal DCSs. Examples of finite sets of deductively closed conditionals are exhibited including many, unlike Boolean algebra, which are finite and yet non-principal, that is, not generated by any one conditional. An introductory section carefully provides motivations and a complete algebraic characterization of the four chosen operations on conditional events. Results are applied to the famous penguin problem.

☆ This work was partially done under contract N00014-95-0251 for Dr. Ralph Wachter, of the Office of Naval Research, 800 North Quincy Street, Arlington, VA 22217-5000, USA. Subsequent support is gratefully acknowledged from ONR and SPAWAR under the In-House Laboratory Independent Research Program.

E-mail address: calabres@nosc.mil (P.G. Calabrese).

0020-0255/02/$ - see front matter
PII: S0020-0255(02)00262-1

Keywords: Conditional; Deduction; Uncertainty; Three-valued; Logic; Probability; Deductive relation; Pre-order; Implication

1. Introduction

While uncertainty of events or propositions poses little conceptual difficulty for making deductions, the same cannot be said for uncertain *conditional* events or propositions. For instance, although there is some doubt whether a roll of an ordinary, six-sided die will turn up a number n less than 3, there is no doubt that if the number is less than 3, then it will also be less than 4. Thus the event "$n < 3$" implies the event "$n < 4$" no matter what the probabilities of these two events because the first event is a subset of the second event. That is Boolean deduction plain and simple. And the probability of the conclusion is at least as great as the probability of the premise. For two-valued (equally conditioned) propositions or events, B is deducible from A if and only if the instances of B include those of A.

However when it comes time to deduce one conditional event from a second conditional event, the picture is not so clear. Unlike two-valued events or propositions, conditionals are inherently three-valued [16,17]. A conditional can be true, or false, or it can be inapplicable, that is, its premise can be false. A conditional has two components, not just one, and so deduction becomes much more complicated when uncertain conditionals are concerned.

The search for appropriate deductive relations between conditionals leads to the following four *elementary* deductive relations between a premise conditional (a|b) and a conclusion conditional (c|d):

(1) "b is a subset of d",
(2) "a and b" is a subset of "c and d",
(3) "a or not b" is a subset of "c or not d", and
(4) "d is a subset of b".

By choosing sub-collections of these four properties, 13 distinct deductive relations are determined.

Although it has become standard in logic since the publication of the Principia Mathematica [31] to routinely reduce a conditional such as "if B, then A" to the non-conditional "either A or not B" – the so-called material conditional, this reduction is not adequate when the events or propositions are allowed to have probabilities between 0 and 1. This inadequacy has been pointed out many times by various authors, among them Adams [4], Calabrese [8], and Lewis [26]. The unconditioned (or universally conditioned) proposition "A or not B" can easily have probability near 1 while the original conditional "if B, then A" has conditional probability near 0. (This problem does not arise when "if B, then A" and "A or not B" are considered wholly true and set equal to the universal event for purposes of deduction and called the "material implication" of A by B.)

Due to the so-called triviality result of Lewis [26], it seemed for a while that there was no way to resolve this discrepancy, and in fact there is none within the standard Boolean algebraic framework of propositions or events. Instead, one must expand the usual Boolean algebra of events (or propositions) to a new algebraic system of event fractions (a|b), ordered pairs of events (or propositions) allowing the usual "and" (conjunction), "or" (disjunction) and "not" (negation) operators, as well as the additional operator "given" (|).

Adams [2–4] had early identified his three so-called "quasi-" operations on conditionals, called quasi merely because the conjunction did not always imply each of the component conditionals and because the disjunction was not always implied by each of the component conditionals. Adams passed over as unsuitable a fourth, "intuitively plausible" operation for iterated conditionals [4, p. 33], apparently because he did not make a distinction between a conditional and an implication. A conditional is not an implication; its a proposition or an event in a given context.

Schay [32] defined two systems of conditionals, one using Adam's quasi-conjunction with a different disjunction and the other using quasi-disjunction with a different conjunction. Others too have identified some of the operations promoted here, including Soboeinski [28,35] in 1952, who chose the first three operations but a very different fourth operation, again meant as an implication. As mentioned by Gilio and Scozzafava [19], in 1985 Bruno and Gilio also defined a system using the quasi-disjunction operation with an alternative conjunction operation.

In 1987 a complete algebraic development, but without deduction, was supplied by Calabrese [9] allowing uncertain conditional events or propositions to be combined in a way faithful to both conditional logic and conditional probability. The object (a|b) can represent "a in the context of b" for logical purposes and also have the conditional probability $P(\text{a}|\text{b}) = P(\text{a and b})/P(\text{b})$ when the conditional is uncertain. (Section 2.3 provides a careful motivation designed to convince skeptics of the correctness of the four chosen operations, and to provide an algebraic characterization of this system of conditional events.)

Subsequently, Goodman and Nguyen published an equivalent formulation [20] but with different operations on the ordered pairs of events based on so-called "first principles" – preservation of as many properties of Boolean algebra as possible. Since conditionals behave in non-Boolean ways, these operations have been hard to apply, although all this has served to highlight the overall algebraic development.

Together with Walker, Goodman and Nguyen served as editors and contributors of a book [21] on conditionals to which the present author contributed a paper [11] on deduction with conditionals. These developments culminated in the publication in December 1994 of a special issue of the IEEE Transactions of Systems, Man and Cybernetics edited by Dubois, Goodman, and Calabrese and devoted to conditional event algebra. As part of this special issue, Dubois and Prade [18], in a virtuoso performance, contrasted the three Goodman–Nguyen–Walker operations with the three Adams–Sobocinski–Calabrese operations and pointed out how these essential differences arise because of the two very different interpretations of the third truth value – "inapplicable" versus "unknown". Using the Adams–Sobocinski–Calabrese operations together with the "probabilistically monotonic" deductive relation previously identified in [10], Dubois and Prade also defined a *conditional knowledge base* and define monotonic deduction from it by adapting Adam's methods to a non-probabilistic context. This development by Dubois and Prade is an important special case of the systems explored in this paper.

Throughout these efforts, one important goal has been the ability in practical situations to combine, deduce and infer using uncertain conditional information without distorting that information. In this regard a large number of examples [14] have been developed and exposited demonstrating the plausibility and applicability of the author's development of conditional event algebra. This has also been corroborated by other researchers including Rödder [29] and Tyszkiewicz [15].

Nevertheless, when attempts were later made to write a computer program [1] to actually perform these operations on even simple events, the inherent complexity of these situations became all too apparent. The computer program got badly bogged down. In retrospect this was not surprising since checking even a simple deduction between two compound Boolean events A = (a_1 or a_2 or a_3 or a_4 or ...) and B = (b_1 or b_2 or b_3 or b_4 or ...), where the a's and b's are themselves Boolean events, requires one to check to see if each instance within each event a_i of A is included in at least one event b_j of B.

This kind of processing has long been recognized as important to logic and computation, having been formalized in [1,24,27,37]. (See, for instance [22], The Mixed Powerdomain, for a summary and tutorial.)

The above expressions for A and B arise quite naturally in the so-called situation space for modeling practical problems: We typically describe a situation by defining a finite number of variables $v_1, v_2, v_3, \ldots, v_n$ each having a finite number of possible values. Even if the variables are all two-valued, they give rise to 2^n different possible complete assignments of values to the variables, and these are just the atoms of the Boolean algebra generated by the original variables. The events of the Boolean algebra generated by these atoms consist

[1] The Multi-Process Algebra Data tool (MuPAD) environment developed at the University of Paderborn, Germany was used to implement the algebra of conditional propositions to produce the Conditional Proposition and Event Processor (CPEP).

of all possible disjunctions (unions) of these 2^n atoms and the number of these events is 2 raised to the power 2^n. So just six initial two-valued variables generate a Boolean algebra with 2^{64} different possible events. A specific event, A, will then typically be expressed as a disjunction of atoms or of subsumed events $a_1, a_2, a_3, \ldots, a_n$. For instance, in a die roll the event of rolling neither 3 nor 4 might be expressed as "<3" or ">4" instead of the longer so-called disjunctive normal form: "1" or "2" or "5" or "6". Even for a small number of variables, brute force computer processing becomes intractable before ever getting to conditionals or probabilities.

Another important class of examples arises from standard track files on detected objects produced by radar and other surveillance systems. These track files consist of records (sequences) of data fields including a time field. If the fields of a record are completely filled in, the record is atomic. These instances (completely filled-in records) are the atoms of the Boolean algebra of subsets of such database atoms under union, intersection and complementation.

A record with missing data in, say one field, can be characterized by the subset of all atomic records that have all but possibly that one field filled with the same values as the given record. In this way a general track file record (possibly having empty fields) is characterized by that subset of atomic records which can consistently complete it. Thus a partially complete record is just a proposition or event, a subset of atomic records. It follows from the above that a sequence (as new records arrive each scan of a sensor), or a collection of such partially complete track file records, must be a second-order predicate – a collection of propositions or events.

Data records in general are usually incomplete and sometimes inconsistent. Thus a general theory of their temporal updating involves at least second-order predicates, subsets of subsets of completely filled data records. The number of such subsets becomes too large for practical computing purposes. So researchers [6,7,27,34] have sought a smaller but still adequately expressive collection of subsets and second-order predicates by which to approximate above and below in some non-deterministic sense any partially complete record. However, these efforts do not appear to have succeeded in producing a practical computation method.

An alternate approach to Bayesian computation now becoming better known utilizes the concept of information entropy [25,33], to defeat complexity by assuming (conditional) independence between variables describing a situation unless some dependence is explicitly known or assumed. Utilizing this and subsequent developments, Rödder [29] and his colleagues at Fern University have constructed a very impressive interactive computer program, with acronym SPIRIT, that can quickly calculate the most likely probability distribution given initial conditional probabilities for some of the relevant conditional events. Professor Rödder has also shown [29,30] that his methods of calculation are consistent with the author's algebra of conditionals and with the deductive relations of conditional event algebra. [2]

The focus here is to explore and further develop the theory of deductive relations (preorders) on conditionals including the deductively closed sets (DCSs) of conditionals that they generate. It should be possible to routinely deduce and infer with uncertain conditional information.

Section 2 on Conditional Event Algebra lays out the algebraic development of conditionals including the four operations and then uses an indicator function representation of conditionals to provide justifications and motivations for the choice of operations and a complete algebraic characterization. Section 3.1 defines the concept of an implication or deductive relation on conditionals and the notion of a DCS of conditionals with respect to an implication relation. In Section 3.2 several different implications are identified based on various Boolean deduction formulas, which are no longer equivalent in the realm of conditionals. Equivalencies between them are exhibited and those remaining are organized into a hierarchy of deductive relations in Fig. 1 of Theorem 3.2.10. Sections 3.3 and 3.4 are essentially new results describing how in general to determine the DCSs of a conditional event algebra with respect to the identified implication relations. Some non-elementary examples of DCSs end the section and illustrate how to generate many other non-elementary examples. This section ends with a solution of the famous penguin problem. Section 4 provides a compendium of all DCSs of the simplest conditional event algebras with respect to all of the identified deductive relations.

2. Conditional event algebra (CEA)

Although George Boole, the recognized father of Boolean algebra, incorporated fractions of propositions in his pioneering work [5], the parallel development of abstract algebra at the time had not proceeded far enough for Boole's immediate followers to make much sense of his far-seeing ideas about division of propositions in regard to conditioning. Consequently, those who pursued his research topics decided to close his system with just his logical counterparts to addition, multiplication and negation, and to call the result "Boolean" algebra. But Boole himself also had the division operation! Boole was explicitly trying to incorporate probability as well as logic into his system.

Conditionals are to logic as fractions are to arithmetic, and the extension is no less dramatic. As the system of integer fractions extends the system of whole

[2] A full description of this entropy approach as applied to conditional event algebra and deduction will be given in a separate paper.

numbers, so also does the system of conditionals (ordered pairs of propositions) extend the underlying Boolean algebra of propositions or probabilistic events allowing the conditional probability $P(\mathrm{a}|\mathrm{b})$ to be the probability of the logical conditional (a|b). This algebra has been rigorously formulated in [9–13]. A brief review of this development follows.

2.1. Formulation of the conditional event algebra $(\mathscr{B}|\mathscr{B})$ of a Boolean algebra $\mathscr{B}$

Start with an initial Boolean algebra, $\mathscr{B}$, of propositions or events such as
(1) all events generated by a surveillance track file or files,
(2) the Boolean algebra generated by any finite or infinite set of propositions,
(3) the Boolean algebra of subsets of a probability sample space Ω, or
(4) {All 64 subsets of the six-element sample space {1, 2, 3, 4, 5, 6}}.

The Boolean set operation "and" is represented by either $\cap$ or $\wedge$. The Boolean operation "or" is represented by $\cup$ or $\vee$. "not" is represented by $'$ or $\neg$.

$(\mathscr{B}|\mathscr{B})$ or $(\mathscr{B}/\mathscr{B})$ will denote the set of ordered pairs, $\{(\mathrm{a}|\mathrm{b}) : \mathrm{a}, \mathrm{b} \text{ in } \mathscr{B}\}$, called the set of *conditionals*, "a given b", of $\mathscr{B}$. The proposition or event "b" is called the *condition*, *premise* or *antecedent* and the proposition or event "a" is called the *consequent* or *conclusion*.

Just as two Boolean propositions or events may be equivalent, that is, refer to the same set of occurrences, so also may two conditionals be equivalent. Analogously, two integer fractions may be equal but look different.

Definition 2.1.1 (*Equivalent conditionals*). Two conditional statements (a|b) and (c|d) are equivalent (=) provided:
(1) their conditions, b and d, are equivalent propositions or events, and
(2) their conclusions, a and c, are equivalent when their common condition is true. In symbols,

$$(\mathrm{a}|\mathrm{b}) = (\mathrm{c}|\mathrm{d}) \quad \text{provided that } \mathrm{b} = \mathrm{d} \text{ and } \mathrm{ab} = \mathrm{cd},$$

where juxtaposition, "ab", denotes "a $\wedge$ b", that is, "a and b".

In other words, two conditionals are equivalent when they have equivalent premises and their conclusions are equivalent assuming that common premise.

This equivalence relation on conditionals implies that for all propositions a and b,

$$(\mathrm{a}|\mathrm{b}) = (\mathrm{ab}|\mathrm{b})$$

and also that for all $\mathrm{a} \in \mathscr{B}$, $(1|0) = (\mathrm{a}|0) = (0|0)$. $(1|0)$ is the "inapplicable" or "undefined" conditional and is denoted U. The Boolean 1 and 0 propositions are represented by $(1|1)$ and $(0|1)$, respectively.

Thus, in any instance a conditional (a|b) can have any one of three truth values:

(a|b) is *true* if (a|b) = (1|1) = 1, i.e., if a is true and b is true

(a|b) is *false* if (a|b) = (0|1) = 0, i.e., if a is false and b is true,

(a|b) is *inapplicable* if (a|b) = (1|0) = U, i.e., if b is false,

Thus, (a|b) is *true* on $a \wedge b$, *false* on $a' \wedge b$, and *inapplicable* on b′.

Note also that "not true" means "false or inapplicable"; "not false" means "true or inapplicable"; "not inapplicable" means "true or false". CEA provides clear distinctions in terminology. For example, "if b, then a" is "not false" on the instances of $a \vee b'$, but it is "true" on the smaller set of instances of $a \wedge b$. No such distinction is available in Boolean algebra.

Having defined conditional events, the *probability* of the truth of a conditional event (a|b) given the truth of the premise of the conditional is defined to be the usual conditional probability $P(a|b) = P(a \text{ and } b)/P(b)$.

2.2. Operations on conditionals

Each of the three operations defined below agrees with the corresponding Boolean operation when applied to conditionals with equivalent conditions. Therefore they extend the Boolean operations.

2.2.1. Relative negation (′)

The relative negation of "a given b" is the "negation of a, given b". That is,

$$(a|b)' = (a'|b).$$

Note that the latter has conditional probability $P(a' \wedge b)/P(b) = [P(b) - P(a \wedge b)]/P(b) = 1 - P(a|b)$.

2.2.2. Disjunction (or)

Concerning disjunction, "If b, then a, or if d, then c" means "If either conditional is applicable, then at least one is true". That is,

$$(a|b) \vee (c|d) = (ab \vee cd)|(b \vee d).$$

To avoid parentheses, the conditioning operator (|) will be assigned an operator preference below disjunction (∨). So negation (′) is first in operator preference, conjunction (∧ or juxtaposition) is second, disjunction is third, and conditioning fourth. So the latter conditional may be written as $(ab \vee cd|b \vee d)$.

2.2.3. Conjunction (and)

Concerning conjunction, "If b, then a and if d, then c" means "if either conditional is applicable, then one is true while the other is not false". That is,

$$\begin{aligned}(a|b) \wedge (c|d) &= [ab(c \vee d') \vee (a \vee b')cd]|(b \vee d)\\ &= (abd' \vee abcd \vee b'cd)|(b \vee d),\end{aligned}$$

which also means "if either conditional is applicable, then either they are both true or else one is true while the other is inapplicable." The latter formula also follows from the standard De Morgan formulas in Boolean algebra relating conjunction and negation to disjunction and negation.

The algebra $(\mathscr{B}|\mathscr{B})$ of conditionals includes the original Boolean algebra $\mathscr{B}$ as those conditionals $(\mathscr{B}|\Omega)$, where Ω is the universal event. In logical notation these are the conditionals $(\mathscr{B}|1)$ whose condition is certain. Analogously, these are like the integer fractions whose denominators are 1. The proposition a is identified with the conditional (a|1). Fixing the condition b yields a Boolean algebra $(\mathscr{B}|\mathrm{b})$, also denoted $(\mathscr{B}/\mathrm{b})$.

These operations corresponding to "not", "and" and "or" allow the usual manipulations, although the resulting system is not wholly Boolean.

2.2.4. Iterated conditioning

A conditional (c|d) may itself be a condition for another proposition or conditional proposition. Due to the largely unrecognized third truth status of conditional statements, to a great degree, natural language is ambiguous about such iterated conditioning. Without additional qualification the iterated form ((a|b)|(c|d)), "(a|b) given (c|d)", could consistently be taken to mean any one of the following:

"a given b and (c|d)" – $(\mathrm{a}|\mathrm{b} \wedge (\mathrm{c}|\mathrm{d}))$,
"a given b and (c|d) are true" – $(\mathrm{a}|\mathrm{b} \wedge \mathrm{c} \wedge \mathrm{d})$,
"(a|b) not false given (c|d) is not false" – $(\mathrm{a} \vee \mathrm{b}'|\mathrm{c} \vee \mathrm{d}')$,
"(a|b) true given (c|d) is not false" – $(\mathrm{a} \wedge \mathrm{b}|\mathrm{c} \vee \mathrm{d}')$,
"(a|b) not false given (c|d) is true" – $(\mathrm{a} \vee \mathrm{b}'|\mathrm{c} \wedge \mathrm{d})$,
"(a|b) true given (c|d) is true" – $(\mathrm{a} \wedge \mathrm{b}|\mathrm{c} \wedge \mathrm{d})$.

Since it can be shown that $\mathrm{b} \wedge (\mathrm{c}|\mathrm{d}) = \mathrm{b} \wedge (\mathrm{c} \vee \mathrm{d}')$, this means that the first possibility (which will be the default interpretation) reduces to

$$(\mathrm{a}|\mathrm{b})|(\mathrm{c}|\mathrm{d}) = (\mathrm{a}|\mathrm{b}(\mathrm{c} \vee \mathrm{d}')).$$

That is, "(a|b) given (c|d)" means "a given that b and (c|d) are not false".

Note that from the above whenever a conditional proposition "if d, then c" is itself a condition, then the corresponding (material conditional) proposition, "either c or else not d", can be used in its place as is commonly done in two-valued logical proof arguments. The conditional (c|d) also acts like its corresponding material conditional, $(\mathrm{c} \vee \mathrm{d}')$, when conjoined ($\wedge$) with any (otherwise unconditioned) proposition b.

2.3. Motivations for the four operations on conditionals

Although intuitive and technical motivations for these operations have been published several times (see for instance [12]), there are many people who still consider them to be debatable. Therefore a restatement and refinement of those motivations is appropriate here. This can be done most efficiently using indicator functions as adopted and employed by Walker [36] to list all possible candidates for operations on conditionals.

2.3.1. Conditionals and indicator functions

It is now well known (see for instance [32, p. 334] or [9, pp. 234–235]) that any conditional (a|b) can be represented as a domain-restricted, measurable indicator function defined on the measurable subset b of Ω as follows:

$$(\mathrm{a}|\mathrm{b})(\omega) = \begin{cases} 1, & \omega \in \mathrm{ab}, \\ 0, & \omega \in \mathrm{a'b}, \\ \mathrm{U}, & \omega \in \mathrm{b'}, \end{cases}$$

U means "undefined". Conversely, any such indicator function assigning 1 and 0, respectively, to disjoint events ab and a′b, and which is undefined elsewhere, determines a unique conditional (a|b).

2.3.2. Boolean functions

By a Boolean function is meant a polynomial built up from the identity function and constant functions on events using negation, conjunction and disjunction a finite number of times.

By the well-known Fundamental Theorem of Boolean algebra such Boolean functions are completely determined by their values, $f(1)$ and $f(0)$, for the two Boolean values 1 and 0. In fact, a Boolean function f of one Boolean variable x is always of the form

$$f(x) = (f(1) \wedge x) \vee (f(0) \wedge x')$$

and a Boolean function f of two variables x and y is always of the form

$$f(x,y) = f(1,1)xy \vee f(1,0)xy' \vee f(1,0)x'y \vee f(0,0)x'y',$$

where for the sake of readability juxtaposition has replaced the conjunction operator in the latter formula.

2.3.3. Operations on conditionals

We restrict attention to operations on conditionals that are defined in such a way that the two components of the image conditional are Boolean functions of the component events of the operands. For instance, the negation operation (′) defined on conditionals in Section 2.2.1 is of the form

$$(\mathrm{a}|\mathrm{b})' = (g(\mathrm{a},\mathrm{b})|h(\mathrm{a},\mathrm{b})),$$

where $g(\mathrm{a},\mathrm{b})$ and $h(\mathrm{a},\mathrm{b})$ are Boolean functions of a and b. The disjunction operation, $\vee$, is of the form

$$(\mathrm{a}|\mathrm{b}) \vee (\mathrm{c}|\mathrm{d}) = (g(\mathrm{a},\mathrm{b},\mathrm{c},\mathrm{d})|h(\mathrm{a},\mathrm{b},\mathrm{c},\mathrm{d})),$$

where g and h are Boolean functions of a, b, c, and d.

Now it is easy to see that any such operation f on conditionals into the set $(\mathscr{B}|\mathscr{B})$ of conditionals determines a unique three-valued truth table by simply setting a and b in turn to the values 1 or 0. If the operation is a one-place function like the negation operation, then the truth table will be:

$(x\|y)$	$f(x\|y)$
$(1\|1)$	$(g(1,1)\|h(1,1))$
$(0\|1)$	$(g(0,1)\|h(0,1))$
$(0\|0)$	$(g(0,0)\|h(0,0))$

It is also known (see [12]) that conversely, any such truth table function k determines a unique operation on conditionals. A one-place truth table function k generates a one-place conditional operation as follows:

$$(k(\mathrm{a}|\mathrm{b}))(\omega) = k((\mathrm{a}|\mathrm{b})(\omega)) = \begin{cases} k(1), & \omega \in \mathrm{ab}, \\ k(0), & \omega \in \mathrm{a'b}, \\ k(\mathrm{U}), & \omega \in \mathrm{b'}. \end{cases}$$

That is, k assigns the measurable indicator function $k((\mathrm{a}|\mathrm{b})(\omega))$ to any three-valued, measurable indicator function $(\mathrm{a}|\mathrm{b})(\omega)$.

A similar statement holds for a two-place, three-valued truth table. A two-place operation like disjunction has a truth table k of the form:

k	1	0	U
1	$k(1,1)$	$k(1,0)$	$k(1,\mathrm{U})$
0	$k(0,1)$	$k(0,0)$	$k(0,\mathrm{U})$
U	$k(\mathrm{U},1)$	$k(\mathrm{U},0)$	$k(\mathrm{U},\mathrm{U})$

Conversely, such a truth table k defines a unique two-place operation f on conditionals by defining its associated indicator function f as follows:

$$f((\mathrm{a}|\mathrm{b}),(\mathrm{c}|\mathrm{d}))(\omega) = k((\mathrm{a}|\mathrm{b})(\omega),(\mathrm{c}|\mathrm{d})(\omega)).$$

2.3.4. *Motivations for the "not", "and" and "or" operations*

It has been shown above that the operations on conditionals are each characterized by a three-valued truth table. The negation operation (′) has a truth table of the form

	Not(′)
1	x
0	y
U	z

In this table $1 = (1|1)$, $0 = (0|1)$ and $U = (0|0)$. Since negation of conditionals is intended to extend the Boolean negation operation, the values x and y must be 0 and 1, respectively. So only z is free. But since the double negation of a conditional should be the conditional back again, z must be U. (If $z = 1$, then $(U')' = 1' = 0 \neq U$; and if $z = 0$, then $(U')' = 0' = 1 \neq U$.) Thus the negation operation can only have the truth table above with $x = 0$, $y = 1$ and $z = U$. So $(a|b)'$ must be $(a'|b)$ since the latter conditional has the same truth table.

Similarly, the conjunction operation ($\wedge$) has a truth table of the form:

$\wedge$	1	0	U
1	1	0	x
0	0	0	y
U	x	y	z

Again, since conjunction of conditionals is intended to extend conjunction of Boolean events, 1's and 0's have been inserted in the table in the appropriate places leaving only five entries undetermined. Since conjunction is intended to be commutative the table must be symmetric about the diagonal. Thus again only three values are still free.

Since conjunction should be idempotent, it follows that $z = U$, and so there are only four possible ways to finish the table.

To motivate the choice of values for x and y, consider how we normally prove a statement A in case B is true or in case its negation B′ is true. We can show that A is true in case B is true and that A is true in case B′ is true, and so therefore prove that A is true. That is, to show A is true, we can show that A is true given B is true and that A is true given that B′ is true. That is, we use that

$$(A|B) \wedge (A|B') = A$$

(We also know that if A is true, then $(A|B)$ and $(A|B')$ will each be true or inapplicable.) Setting $B = 0$ and $A = 1$ in the above equation yields that

$$(1|0) \wedge (1|1) = 1.$$

That is, $U \wedge 1 = 1$. Therefore $x = 1$ in the conjunction table.

Similarly, we can show that A is impossible (0) by showing that both $(A|B)$ is false and $(A|B')$ is false. That is, we use that

$$(0|0) \wedge (0|1) = 0.$$

This also follows by simply setting $A = 0$ instead of $A = 1$ above. So $U \wedge 0 = 0$, and therefore in the conjunction table y must be 0. That completes the conjunction truth table.

(Alternately, consider how a questionnaire with conditional questions is interpreted when some of the conditional questions do not apply to an individual: The answers to questions are basically conjoined and any inapplicable questions are ignored. They do not make the whole set of answers inapplicable or undefined.)

The disjunction operation ($\vee$) has a truth table like

$\vee$	1	0	U
1	1	1	x
0	1	0	y
U	x	y	U

and like the conjunction operation most of the table is determined because the operation is intended to extend Boolean disjunction to conditionals and be commutative and idempotent. These properties will all be satisfied and the table completed by simply specifying that the De Morgan's laws should also hold for conditionals. That is, we require that the negation of a conjunction of conditionals be equivalent to the disjunction of the negations of those conditionals. So since $(0|0) \wedge (0|1) = 0$, taking negations on both sides, yields that $(0|0)' \vee (0|1)' = 0'$, which is equivalent to $(1|0) \vee (1|1) = 1$. That is, $\mathrm{U} \vee 1 = 1$. So in the disjunction table x must be 1. Similarly, since $(1|0) \wedge (1|1) = 1$, taking negations on both sides yields that $\mathrm{U} \vee 0 = 0$. So in the table y must be 0.

We can now summarize the results of this subsection in the following theorem.

2.3.5. Algebraic characterization theorem

The only unary operation ($'$) on conditionals whose double operation is idempotent and that extends Boolean negation on events and whose components are Boolean functions of the components of the operand conditional is that of Section 2.2.1, namely, $(\mathrm{a}|\mathrm{b})' = (\mathrm{a}'|\mathrm{b})$. Furthermore, the only commutative, idempotent binary operations on conditionals that extend the conjunction and disjunction operations on events and whose components are Boolean functions of the components of the operands, and which satisfy the De Morgan's formulas, and which satisfy the property $(\mathrm{c}|\mathrm{d}) \wedge (\mathrm{c}|\mathrm{d}') = (\mathrm{c}|\mathrm{d} \vee \mathrm{d}')$ for all events c and d, are those of Definitions 2.2.2 and 2.2.3, namely, $(\mathrm{a}|\mathrm{b}) \vee (\mathrm{c}|\mathrm{d}) = (\mathrm{ab} \vee \mathrm{cd}|\mathrm{b} \vee \mathrm{d})$ and $(\mathrm{a}|\mathrm{b}) \wedge (\mathrm{c}|\mathrm{d}) = (\mathrm{abd}' \vee \mathrm{abcd} \vee \mathrm{b}'\mathrm{cd}|\mathrm{b} \vee \mathrm{d})$.

2.3.6. Alternate formulations

Adams [4] also defines what he calls "quasi-operations" on conditionals that are equivalent to those of Sections 2.2.2 and 2.2.3 but he does not settle on an iterated conditional operation even though he does identify situations when update information is conditional. Adams refers to his operations as "quasi" specifically because they are not monotonic, that is, for example, the conjunction of conditionals does not necessarily imply each of the conditionals being conjoined. But that is just one way operations on conditionals differ from operations on ordinary events.

Schay [32] defines two 3-operation algebras, the first of which Goodman, Nguyen and Walker erroneously identify as equivalent to the author's first three operations (See p. 6 and p. 92 of [20]). This error was then propagated by Hailperin [23, p. 266]. But the truth is that the author's disjunction operation is in one of Schay's systems and the conjunction operation is in the other of Schay's systems, and so neither of Schay's systems contains the three operations of Sections 2.2.1–2.2.3. No wonder Hailperin, as he says [23, p. 261], gets results so different from those of Schay.

Concerning the third truth-value, Hailperin quotes Schay, who says for conditionals a and b: "... if a is defined and b is undefined, then we put $\max\{a,b\} = a$ and $\min\{a,b\} = a$". Hailperin finds [23, p. 262] it "difficult to conceive of a rationale for defining max and min in this manner". The difficulty is in the misinterpretation of the third truth-value. When a conditional is "undefined" that is nothing like saying that its truth-value is unknown. In the latter situation assigning a truth-value or better, a probability, between 0 and 1 is appropriate. But a conditional with a false condition is not somewhat true; it does not deserve to have a truth-value between 0 and 1 as though it were somewhat true. It is simply inapplicable – a completely different category. Conjoining a true or false statement with one that is inapplicable leaves the applicable statement unchanged. Is that so difficult to conceive? Is that not what we do when we skip an inapplicable question on a questionnaire and fill out the other questions? In principle, we expect the reader to conjoin all of our answers, conditional or not, and ignore the inapplicable questions. We do not expect the reader to declare the whole form "undefined" merely because one inapplicable conditional was encountered.

Hailperin also takes issue with the author [23, p. 264] for claiming that when rolling one 6-sided die the statement "if the roll is even, then it will be a 6 or if the roll is odd, then it will be a 5" is intuitively equivalent to the statement "The roll will be a 6 or a 5". He does not see it as so clear cut and apparently finds the Goodman–Nguyen alternative translation as "If the roll is 6 or 5, then the roll will be 6 or 5", to be intuitively preferable, even though it has a conditional probability of 1.

Shifting to the conjunction operation in keeping with the motivations presented in this paper, one wonders whether Hailperin finds the statement "If the roll is even, then it will be a 5 or a 6 and if the roll is odd, then it will be a 5 or a 6" to be intuitively equivalent to "the roll will be a 5 or a 6 whether or not it is even". That is, after all, the intuitive meaning of a proof by cases, and it also works when the conclusion is uncertain. But the conjunction operation favored by Hailperin, Goodman and Nguyen has it that the statement is instead equivalent to "if the roll is 1, 2, 3, or 4, then falsity". That might be an acceptable translation for some logical purposes but it has conditional probably 0 instead of 2/6, which is the intuitive probability of getting a 5 or 6 in one roll of a die given that the roll is odd or even.

2.3.7. Motivations for the iterated conditioning operation

The fourth operation on conditionals, defined in Section 2.2.4, is essential for closure of the algebraic system. Without it, conditioning cannot be performed on conditionals, and deduction or inference of a conditional by another conditional remains out of reach of the syntax of the algebra. The ability of the system to express other operations on conditionals is also greatly expanded by the inclusion of the fourth operation. As shown by Tyszkiewicz [15] and his co-authors, the other operations on conditionals defined by Goodman et al. [20] can all be expressed in terms of the four operations defined in Sections 2.2.1–2.2.4.

When discussing the author's system of conditionals some authors [36, p. 1706], have ignored the iterated conditioning operation as though it were a thing apart, preferring to characterize the system in terms of just the first three operations and identifying it with the first three operations of the system of Sobocinski [28,35]. While the first three operations of Sobocinski are equivalent to those of Sections 2.2.1–2.2.3, the fourth is quite different from Sobocinski's implication operation.

When the fourth operation is included it is clear that the operations of Sections 2.2.1–2.2.4 are not a repetition of those that have previously been explored in the literature. (This is also apparent when it is noted that an iterated conditional $((a|b)|(c|d))$ is not interpreted to be an implication $(c|d) \rightarrow (a|b)$, as is done by most authors. Nor is the conditional itself an implication. Rather, implications are separately defined as in this paper.)

Now the most straightforward way to motivate the iterated conditional operation of Section 2.2.4 is to extend the following rule for unconditional (Boolean) events a, b and c:

$$((a|b)|c) = (a|b \wedge c).$$

Here again, mathematicians routinely prove theorems by successive conditioning according to the above formula. We very often read arguments of the form "if c is true, then if also b is true, then a will be true". The proof will then proceed to show that if both b and c are true, then a will be true, and no one will dispute the matter.

In this regard, Adams [4, p. 33] mentions this iterated conditional simplification as intuitively plausible but says that it would entail "giving up modus ponens in application to conditionals with conditional consequents". Adams' example is that $((A|B)|A)$ would be certain and so we should always infer $(A|B)$ from A, but, he says, we know independently that this inference is not always sound. However there seems to be no real problem with this inference when we are talking about conditionals instead of implications. When A is true the conditional $(A|B)$ will be true as long as B is true, or inapplicable if B is false. We are not inferring that $(A|B)$ is always true.

The general iterated conditional ((a|b)|(c|d)) can then be reduced using this formula as follows:

$$((a|b)|(c|d)) = (a|(b \wedge (c|d))),$$

which can be further reduced to $(a|(b \wedge (c \vee d')))$, which is a conditional with Boolean components as required.

The truth table for this conditioning operation, |, is:

(|)	1	0	U
1	1	U	1
0	0	U	0
U	U	U	U

where the first two columns extend the case of iterated Boolean (unconditioned) events, and the third column expresses the interpretation that an undefined condition leaves the consequent unchanged. That is, $(1|U) = 1$, $(0|U) = 0$ and $(U|U) = U$.

Adding to the premise of the characterization theorem of Section 2.3.5 the iterated conditioning rule

$$((a|b)|c) = (a|bc)$$

for any events or conditional events a, b and c, the whole system of four operations on conditionals (Sections 2.2.1–2.2.4) has been algebraically characterized.

2.4. *Deduction of conditionals by conditionals*

If statements of the certainty of a proposition or conditional proposition are allowed such as "c is never false" or "(c|d) is never false", then there are many more forms of deduction involving the equivalence relation "=". It becomes clear, then, that deduction and inference involving conditionals requires special care in specifying exactly what is being assumed (the conditions), and secondly exactly what is being deduced or inferred, and thirdly exactly what type of deduction between conditionals is being invoked.

3. Deductive relations and deductively closed sets

Due to the existence of four (not just two) propositions between two conditionals, deduction takes several forms. To put them into a common context the following algebraic definitions are useful:

3.1. General deductive relations on conditionals

First some basic definitions and a theorem set the stage:

Definition 3.1.1. A *deductive relation* (also called a *preorder* or an *implication*) is a reflexive and transitive relation, $\leq$, on a set.

The symbol "$\leq$" is used to denote deduction instead of an arrow, $\Rightarrow$ or $\rightarrow$, in order to connote the interpretation of deduction in terms of inclusion: In Boolean logic, an event A implies a second event B if and only if every instance ω in A is also in B; that is, every occurrence of A is an occurrence of B. So event A implies event B if and only if A is a subset of B.

Definition 3.1.2. A deductive relation $\leq$ for conditionals has the *Boolean Extension Property* if and only if for all propositions a, b, and c,

$$\mathrm{ab} \leq \mathrm{cb} \quad \text{implies } (\mathrm{a}|\mathrm{b}) \leq (\mathrm{c}|\mathrm{b}).$$

That is, a deductive relation $\leq$ on $\mathscr{B}/\mathscr{B}$ has the Boolean Extension Property if it extends the Boolean deduction relation $\leq$ of every Boolean sub-algebra $\mathscr{B}$/b, of $\mathscr{B}/\mathscr{B}$, that is, if it agrees with Boolean deduction on every Boolean sub-algebra $\mathscr{B}$/b, of $\mathscr{B}/\mathscr{B}$.

Definition 3.1.3. The deductive relation $\leq$ is *well-defined* with respect to equality $(=)$ of conditionals if and only if whenever $(\mathrm{a}|\mathrm{b}) = (\mathrm{a}_1|\mathrm{b}_1)$ and $(\mathrm{c}|\mathrm{d}) = (\mathrm{c}_1|\mathrm{d}_1)$ and $(\mathrm{a}|\mathrm{b}) \leq (\mathrm{c}|\mathrm{d})$ then $(\mathrm{a}_1|\mathrm{b}_1) \leq (\mathrm{c}_1|\mathrm{d}_1)$.

Theorem 3.1.4. *If a deductive relation $\leq$ has the Boolean Extension Property then it is well-defined with respect to equality $(=)$ of conditionals.*

Proof of Theorem 3.1.4. Suppose $(\mathrm{a}|\mathrm{b}) = (\mathrm{a}_1|\mathrm{b}_1)$ and $(\mathrm{c}|\mathrm{d}) = (\mathrm{c}_1|\mathrm{d}_1)$ and $(\mathrm{a}|\mathrm{b}) \leq (\mathrm{c}|\mathrm{d})$. By the definition of equality $(=)$ of conditionals, $\mathrm{b} = \mathrm{b}_1$ and $\mathrm{d} = \mathrm{d}_1$, and $\mathrm{ab} = \mathrm{a}_1\mathrm{b}_1 = \mathrm{a}_1\mathrm{b}$ and similarly $\mathrm{cd} = \mathrm{c}_1\mathrm{d}$. So $(\mathrm{a}_1|\mathrm{b}_1) = (\mathrm{a}_1|\mathrm{b}) \leq (\mathrm{a}|\mathrm{b}) \leq (\mathrm{c}|\mathrm{d}) = (\mathrm{c}|\mathrm{d}_1) \leq (\mathrm{c}_1|\mathrm{d}_1)$ using both substitution and the Boolean Extension Property twice. By transitivity of $\leq$ it follows that $(\mathrm{a}_1|\mathrm{b}_1) \leq (\mathrm{c}_1|\mathrm{d}_1)$. $\square$

According to one standard definition, a subset S of *unconditioned* events (or propositions) is a deductively closed set of events (or propositions) provided that: (1) the conjunction of any two propositions in S is also in S, and (2) any event that subsumes an event in S is also in S. A similar definition also works for deductively closed subsets of conditional propositions (or events) with respect to some specified deductive relation (preorder):

Definition 3.1.5 (*Deductively closed sets*). A subset H of $\mathscr{B}/\mathscr{B}$ is said to be *deductively closed* with respect to a deductive relation $\leq_x$ if and only if H has both of the following properties:

(1) If $(a|b) \in H$ and $(c|d) \in H$, then $(a|b) \wedge (c|d) \in H$, and

(2) If $(a|b) \in H$ and $(a|b) \leq_x (c|d)$, then $(c|d) \in H$.

The first property will be called *conjunctive closure* and the second will be called *deductive closure*. H is said to be a *deductively closed set* (DCS) of conditionals with respect to $\leq_x$. For Boolean propositions, $\leq_x$ can be the standard deduction relation.

Definition 3.1.6 (*Deductive equivalence* $(=_x)$ *of conditionals*)

$$(a|b) =_x (c|d) \quad \text{if and only if } (a|b) \leq_x (c|d) \text{ and } (c|d) \leq_x (a|b).$$

3.2. Extensions of Boolean implication

The following definitions are natural since they are equivalent in Boolean algebra but not so in the algebra of conditionals $\mathscr{B}/\mathscr{B}$.

Definition 3.2.1 (*Conjunctive implication* $(\leq_\wedge)$)

$$(a|b) \leq_\wedge (c|d) \quad \text{if and only if } (a|b) \wedge (c|d) = (a|b).$$

Definition 3.2.2 (*Disjunctive implication* $(\leq_\vee)$)

$$(a|b) \leq_\vee (c|d) \quad \text{if and only if } (a|b) \vee (c|d) = (c|d).$$

Definition 3.2.3 (*Probabilistically monotonic implication* $(\leq_{\mathrm{pm}})$)

$$(a|b) \leq_{\mathrm{pm}} (c|d) \quad \text{if and only if } (c|d) \vee (a|b)' = (1|d \vee b).$$

Proving reflexivity and transitivity for the above three deductive relations is trivial except perhaps transitivity for the last one, which will be proved as a corollary to Theorem 3.2.10. Similarly, showing the following three plausible implication relations are reflexive and transitive is an easy corollary of Theorems 3.2.7 and 3.2.8, which prove that they are all equivalent to each other.

Definition 3.2.4 (*Non-falsity implication* $(\leq_{\mathrm{nf}})$)

$$(a|b) \leq_{\mathrm{nf}} (c|d) \quad \text{if and only if } (a \vee b') \leq (c \vee d').$$

That is, if $(a|b)$ is not false, then $(c|d)$ is not false.

Definition 3.2.5 (*Necessary implication* ($\leq_n$))

$$(a|b) \leq_n (c|d) \quad \text{means if } (a|b) = (1|b), \text{ then } (c|d) = (1|d).$$

That is, if a is necessary given b, then c is necessary given d.

Definition 3.2.6 (*Conditional necessity implication* ($\leq_c$))

$$(a|b) \leq_c (c|d) \quad \text{means } (c|d)|(a|b) = (1|d)|(a|b).$$

That is, given (a|b), c is necessary given d.

Theorem 3.2.7. *Non-falsity implication is equivalent to conditional necessity implication.*

Proof of Theorem 3.2.7. $(c|d)|(a|b) = (1|d)|(a|b)$ iff $[(c|d) = (1|d)]|(a|b)$ iff $(d \leq c)|(a|b)$ iff $(d|(a|b)) \leq (c|(a|b))$ iff $d(a \vee b') \leq c(a \vee b')$ iff $dc'(a \vee b') \leq 0$ iff $(a \vee b') \leq (c \vee d')$. (The result also follows by simplifying $(c|d)|(a|b)$ to $(c|d(a \vee b'))$ and $(1|d)|(a|b)$ to $(1|d(a \vee b'))$, which are equal. So $cd(a \vee b') = d(a \vee b')$, which is equivalent to $(a \vee b') \leq (c \vee d')$.) □

Theorem 3.2.8. *Non-falsity implication is equivalent to necessary implication.*

Proof of Theorem 3.2.8. $(a|b) = (1|b)$ means $ab = b$, which means $b \leq a$, which means $(a \vee b') = 1$. Similarly $(c|d) = (1|d)$ is equivalent to $(c \vee d') = 1$. So the necessary implication $(a|b) \leq_n (c|d)$ means that $(a \vee b') = 1$ implies $(c \vee d') = 1$, which can only be true providing $a \vee b' \leq c \vee d'$. □

The last assertion needs some proof:

From Definition 3.2.4, for which transitivity is trivial, and Theorems 3.2.7 and 3.2.8, it follows that all three relations of Definitions 3.2.4–3.2.6 are transitive.

Theorem 3.2.9 (The certainty of non-falsity). *If whenever $a \vee b' = 1$ then $c \vee d' = 1$, then $a \vee b' \leq c \vee d'$. (The converse is trivial.)*

Proof of Theorem 3.2.9. Suppose e is an arbitrary proposition. By hypothesis, if $(a \vee b'|e) = (1|e)$, then $(c \vee d'|e) = (1|e)$. So for all propositions e, $[(a \vee b')e = e]$ implies $[(c \vee d')e = e]$. That is, for all propositions e, $[e \leq (a \vee b')]$ implies $[e \leq (c \vee d')]$. Setting $e = (a \vee b')$ yields that $[(a \vee b') \leq (a \vee b')]$ implies $[(a \vee b') \leq (c \vee d')]$. Since $[(a \vee b') \leq (a \vee b')]$ is always true, so too must $[(a \vee b') \leq (c \vee d')]$. □

Since these three relations are equivalent, it suffices to show any one of them to be reflexive and transitive. The non-falsity relation of Definition 3.2.4 is obviously reflexive and transitive.

Theorem 3.2.10. *The deductive relations* $\leq_\wedge, \leq_\vee, \leq_{pm}$, *and* $\leq_{nf}$ *on conditionals defined by the following equations can be reduced to the Boolean relations listed on the right-hand side*:

$\leq_\wedge$: $(a|b) \wedge (c|d) = (a|b)$ *if and only if* $(a \vee b') \leq (c \vee d')$ *and* $(b' \leq d')$.
$\leq_\vee$: $(a|b) \vee (c|d) = (c|d)$ *if and only if* $(a \wedge b) \leq (c \wedge d)$ *and* $(b \leq d)$.
$\leq_{pm}$: $(a|b)' \vee (c|d) = (1|b \vee d)$ *iff* $(a \vee b') \leq (c \vee d')$ *and* $(a \wedge b) \leq (c \wedge d)$.
$\leq_{nf}$: $[(c|d)|(a|b)] = [(1|d)|(a|b)]$ *if and only if* $(a \vee b') \leq (c \vee d')$.

Proof of Theorem 3.2.10. Concerning the first equation of the theorem, suppose $(a|b) \wedge (c|d) = (a|b)$. Applying the conjunction operation and the definition of equivalence for conditionals yields the two equalities $abd' \vee b'cd \vee abcd = ab$ and $b \vee d = b$. So immediately from just the second equality it follows that $d \leq b$, which is equivalent to $b' \leq d'$. Since $b'd = 0$ the first equality becomes $abd' \vee 0 \vee abcd = ab$, which is equivalent to $ab(d' \vee cd) = ab$. The latter is equivalent to $ab \leq (d' \vee cd)$, and since $b' \leq d'$, it follows immediately that $b' \vee ab \leq (d' \vee cd)$, which is $(a \vee b') \leq (c \vee d')$. Reversing these steps produces the converse. For the second equality, applying the disjunction operation and the definition of equality of conditionals yields that $ab \vee cd = cd$ and $b \vee d = d$, which are equivalent to $ab \leq cd$ and $b \leq d$, respectively. Reversing these steps yields the converse of the second equation. Concerning the third equality of the theorem, applying negation and disjunction for conditionals and the definition of equivalence of conditionals yields $a'b \vee cd = b \vee d$ and $b \vee d = b \vee d$. Conjunction of both sides of the first equality by ab yields $abcd = ab$, which is equivalent to $ab \leq cd$. Conjunction of both sides of the first equality instead by $c'd$ yields $(a'b)(c'd) = (c'd)b \vee c'd = c'd$. So $c'd \leq a'b$, which, by taking complements of both sides and reversing the inequality, is equivalent to $(a \vee b') \leq (c \vee d')$. The converse of the third equality of the theorem follows since if $ab \leq cd$ and $c'd \leq a'b$, then $a'b \vee cd = (a'b \vee c'd) \vee (cd \vee ab) = (a'b \vee ab) \vee (cd \vee c'd) = b \vee d$. That is $a'b \vee cd = b \vee d$, which is equivalent to $(a|b)' \vee (c|d) = (1|b \vee d)$. Finally, concerning the fourth equality of the theorem, applying the conditioning operation and the definition of equivalence of conditionals yields that $cd(a \vee b') = d(a \vee b')$. Disjunction on both sides of the latter equality by $d'(a \vee b')$ yields $(cd \vee d')(a \vee b') = (a \vee b')$, which is equivalent to $(a \vee b') \leq (c \vee d')$. Conversely, if $(a \vee b') \leq (c \vee d')$ then conjunction of both sides by d yields that $(a \vee b')d \leq cd$. So $(cd)(a \vee b')d = (a \vee b')d$. That is $(cd)(a \vee b') = (a \vee b')d$, which is equivalent to the left-hand side of the fourth equality of the theorem. That completes the proof of Theorem 3.2.10. □

The reduction of the relations listed in Theorem 3.2.10 to their associated Boolean relations also exhibits the obvious transitivity of those relations.

This theorem also suggests that the Boolean relations on the right-hand sides of "if and only if" in Theorem 3.2.10 define *elementary implications* on conditionals. Indeed, there is the following hierarchy of implications (Fig. 1):

Trivial implications	Elementary implications
1 – Implication of identity ($\leq_1$)	tr – Implication of truth ($\leq_{tr}$)
(a|b) $\leq_1$ (c|d) iff (a|b) = (c|d)	(a|b) $\leq_{tr}$ (c|d) iff ab $\leq$ cd
bo – Boolean deduction ($\mathscr{B}$| fixed b) ($\leq_{bo}$)	nf – Implication of non-falsity ($\leq_{nf}$)
(a|b) $\leq_{bo}$ (c|d) iff b = d and ab $\leq$ cd	(a|b) $\leq_{nf}$ (c|d) iff (a $\vee$ b′) $\leq_{nf}$ (c $\vee$ d′)
ec – Implication of equal conditions ($\leq_{ec}$)	ap – Implication of applicability ($\leq_{ap}$)
(a|b) $\leq_{ec}$ (c|d) iff b = d	(a|b) $\leq_{ap}$ (c|d) iff b $\leq$ d
0 – Universal implication	ip – Implication of inapplicability ($\leq_{ip}$)
(a|b) $\leq_0$ (c|d) for all (a|b) & (c|d)	(a|b) $\leq_{ip}$ (c|d) iff d $\leq$ b
Two elementaries combined	**Three elementaries combined**
$\vee$ – Disjunctive implication ($\leq\vee$)	m$\vee$ – (Probabilistically) monotonic and applicability implication ($\leq_{m\vee}$)
pm – Probabilistically monotonic implication; ($\leq_{pm}$)	m$\wedge$ – (Probabilistically) monotonic and inapplicability implication ($\leq_{m\wedge}$)
$\wedge$ – Conjunctive implication ($\leq_\wedge$)	

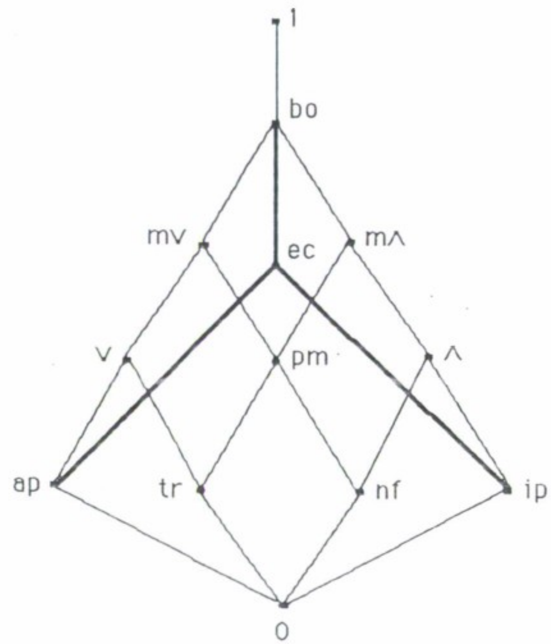

Fig. 1. Hierarchy of implications (deductive relations) for conditionals.

Note also that all four elementary preorders have the Boolean extension property because for $x \in \{\text{ap}, \text{ip}\}$, the relation $(\text{a}|\text{b}) \leq_x (\text{c}|\text{b})$ holds whether or not $(\text{ab} \leq \text{cb})$ holds. For $x = \text{tr}$, $(\text{a}|\text{b}) \leq_x (\text{c}|\text{b})$ reduces to the hypothesis $\text{ab} \leq \text{cb}$, and finally for $x = \text{nf}$, $(\text{a}|\text{b}) \leq_x (\text{c}|\text{b})$ reduces to $\text{a} \vee \text{b}' \leq \text{c} \vee \text{b}'$, which becomes $\text{ab} \leq \text{cb}$ after conjunction on both sides by b. Since $\text{ab} \leq \text{cb}$ implies $(\text{a}|\text{b}) \leq_x (\text{c}|\text{b})$ for all four elementary preorders x, the other preorders in the hierarchy also satisfy the Boolean extension property. (See also [10, p. 687–688] and [11, pp. 85–100].)

3.2.11. Probability relationships. Probability relationships naturally flow from the above implication relations, although among the 13 defined above only monotonic implication, and those above it in the hierarchy, ensure probabilistic monotonicity. That is, if $(\text{q}|\text{p}) \leq_{\text{pm}} (\text{s}|\text{r})$, then $P(\text{q}|\text{p}) \leq P(\text{s}|\text{r})$. For instance, conditional necessity implication, $(\text{q}|\text{p}) \leq_{\text{c}} (\text{s}|\text{r})$, ensures only that $P(\text{q} \vee \text{p}') \leq P(\text{s} \vee \text{r}')$.

3.2.12. Conditional equivalence. Although the three equivalent implication relations ($\leq_{\text{nf}}, \leq_{\text{c}}$, and $\leq_{\text{n}}$) are reflexive and transitive, they are not anti-symmetric. That is, $(\text{a}|\text{b}) \leq_{\text{nf}} (\text{c}|\text{d})$ and $(\text{c}|\text{d}) \leq_{\text{nf}} (\text{a}|\text{b})$ together do not imply that $(\text{a}|\text{b}) = (\text{c}|\text{d})$. That is, $(\text{a} \vee \text{b}') = (\text{c} \vee \text{d}')$ does not imply that $\text{b} = \text{d}$ and $\text{ab} = \text{cd}$. As such, $\leq_{\text{nf}}$ is a quasi-order (also called a preorder), but not a partial order. Equivalently, $(\text{a}|\text{b}) \leq_{\text{nf}} (\text{c}|\text{d})$ if and only if $\text{c}'\text{d} \leq \text{a}'\text{b}$. The latter means that "if (c|d) is false then (a|b) is false." In these terms, $(\text{a}|\text{b}) =_{\text{nf}} (\text{c}|\text{d})$ means that (a|b) is false if and only if (c|d) is false.

3.2.13. Conditional implication and the contrapositive. Note that a conditional proposition (a|b) and its contrapositive, $(\text{b}'|\text{a}')$, are non-falsely equivalent: $(\text{a}|\text{b}) =_{\text{nf}} (\text{b}'|\text{a}')$. This is reassuring since a conditional proposition and its contrapositive should be logically equivalent when regarded as wholly non-false (as when assumed or conditioned upon) but not equivalent, nor even have the same probability, when regarded as partially false. In fact, a conditional (a|b) is false if and only if its contrapositive $(\text{b}'|\text{a}')$ is false, but if either (a|b) or $(\text{b}'|\text{a}')$ is true the other is inapplicable. They can also both be inapplicable. (See [9, p. 222], for a comparison of the probability of (a|b) with the probability of $(\text{b}'|\text{a}')$.)

3.2.14. Certainty theorem. If, whenever $\text{ab} = 1$ then $\text{cd} = 1$, then $\text{ab} \leq \text{cd}$. That is, if, whenever (a|b) is true then (c|d) is true, then $\text{ab} \leq \text{cd}$. (The converse is easy.)

Proof of Theorem 3.2.14. If for all propositions e, $(\text{ab}|\text{e}) = (1|\text{e})$ implies $(\text{cd}|\text{e}) = (1|\text{e})$, then it follows as above that for all e, $[\text{e} \leq \text{ab}]$ implies $[\text{e} \leq \text{cd}]$. Then setting $\text{e} = \text{ab}$ yields that $\text{ab} \leq \text{cd}$.

So the "necessarily true" preorder $\leq_{\text{nt}}$, defined by:

$$(\text{a}|\text{b}) \leq_{\text{nt}} (\text{c}|\text{d}) \quad \text{means if } (\text{ab} = 1), \text{ then } (\text{cd} = 1)$$

is equivalent to the preorder $\leq_{\text{tr}}$, defined by $(\text{ab} \leq \text{cd})$. $\square$

The deductive relations above and some new ones built up from the elementary deductions have been organized into a hierarchy (Fig. 1) defined completely in terms of Boolean relations on the Boolean components of the conditionals. These results are summarized in the following:

3.2.15. Interpretations of elementary deductive conditionals.
$\leq_{tr}$: $ab \leq cd$ "If (a|b) is true, then (c|d) is true."
$\leq_{nf}$: $a \vee b' \leq c \vee d'$ "If (a|b) is not false, then (c|d) is not false."
$\leq_{ap}$: $b \leq d$ "If (a|b) is applicable, then (c|d) is applicable."
$\leq_{ip}$: $b' \leq d'$ "If (a|b) is inapplicable, then (c|d) is inapplicable."

3.2.16. Two elementary deductive relations combined. From these four elementary deductive relations, three of the previously identified deductive relations ($\leq_{\wedge}$, $\leq_{\vee}$, and $\leq_{pm}$) can be defined by combining the properties of two of the elementary deductive relations:

$$(a|b) \leq_{\wedge} (c|d) \quad \text{if and only if } (a \vee b') \leq (c \vee d') \text{ and } (b' \leq d')$$
(Conjunctive implication).

$$(a|b) \leq_{\vee} (c|d) \quad \text{if and only if } ab \leq cd \text{ and } b \leq d$$
(Disjunctive implication).

$$(a|b) \leq_{pm} (c|d) \quad \text{if and only if } ab \leq cd \text{ and } (a \vee b') \leq (c \vee d')$$
(Probabilistically monotonic implication).

3.2.17. Three elementary deductive relations combined. Two additional deductive relations arise by combining the properties of $\leq_{pm}$ with the property of $\leq_{ap}$, or of $\leq_{ip}$:
$(a|b) \leq_{m\wedge} (c|d)$ if and only if $ab \leq cd$, $(a \vee b') \leq (c \vee d')$, and $b' \leq d'$.
$(a|b) \leq_{m\vee} (c|d)$ if and only if $ab \leq cd$, $(a \vee b') \leq (c \vee d')$, and $b \leq d$.

3.2.18. Deductive equivalence theorem. For $x \in \{ap, tr, nf, ip, ec, 0\}$, $(a|b) =_x (c|d)$ does not imply $(a|b) = (c|d)$, but for the other seven deductive relations in the hierarchy, $(a|b) =_x (c|d)$ implies equality of (a|b) and (c|d) as conditionals.

Proof of Theorem 3.2.18. For $x = 0$, $(a|b) =_x (c|d)$ is true for any two unequal conditionals. So equality of (a|b) and (c|d) is not implied. For $x \in \{ap, ip, ec\}$, $(a|b) =_x (c|d)$ simply implies that (a|b) and (c|d) have common condition. That is, $b = d$, but they need not be equal. For $x = nf$, $(a|b) =_x (c|d)$ implies only $a \vee b' = c \vee d'$, which does not imply equality of (a|b) and (c|d). Similarly for $x = tr$, because $(a|b) =_x (c|d)$ implies $ab = cd$, but not that $b = d$, which is required for equality. For $x = pm$, $(a|b) =_x (c|d)$ implies both $ab = cd$ and $a \vee b' = c \vee d'$. So $a'b = c'd$. Therefore $b = ab \vee a'b = cd \vee c'd = d$. So $(a|b) =_{pm} (c|d)$ implies $(a|b) = (c|d)$. Each of the other two deductive relations $\leq_{m\wedge}$ and $\leq_{m\vee}$ are stronger than $\leq_{pm}$ and so imply equality too. For $x = bo$,

$(a|b) =_x (c|d)$ trivially implies both $b = d$ and $ab = cd$ and so $(a|b) = (c|d)$. For $x = 1$, trivially $(a|b) =_x (c|d)$ means $(a|b) = (c|d)$. Finally the remaining two deductive relations, $\leq_\wedge$ and $\leq_\vee$, are stronger than one of $\leq_{ap}$ or $\leq_{ip}$ and so $(a|b) =_x (c|d)$ implies that $b = d$. But $(a|b) =_\vee (c|d)$ implies $ab = cd$ and thus that $(a|b) = (c|d)$. Similarly, $(a|b) \leq_\wedge (c|d)$ implies $a \vee b' = c \vee d'$. Since $b = d$, $(a \vee b')b = (c \vee d')d$. That is, $ab = cd$, and so $(a|b) = (c|d)$. That completes the proof of Theorem 3.2.18. □

3.3. Construction of deductively closed sets

Having described the hierarchy of deductive relations on conditionals it remains to describe how to construct the deductively closed sets with respect to them.

To show that all of these deductive relations have at least one closed set of conditionals the following are listed: The set $H \doteq \mathscr{B}/\mathscr{B}$, the whole set of conditionals, is deductively closed with respect to any deductive relation $\leq_x$. $H = \{(q|p) : p \leq q\}$ is deductively closed with respect to $\leq_{nf}$. $H = \{(1|1)\}$ is deductively closed with respect to $\leq_\vee$. $H(b) = \{(x|y) : y \leq b\}$ is deductively closed with respect to $\leq_{ip}$. $K(b) = \{(x|y) : b \leq y\}$ is deductively closed with respect to $\leq_{ap}$. $L(b) = \{(x|y) : b \leq xy\}$ is deductively closed with respect to $\leq_{tr}$.

Theorem 3.3.1 (Conjunction theorem for deductively closed sets with respect to two deductive relations). *Suppose that H_x is a deductively closed set of conditionals with respect to deductive relation $\leq_x$, and that H_y is a deductively closed set of conditionals with respect to deductive relation $\leq_y$. Then the intersection $H_x \cap H_y$ is a DCS, $H_{x \cap y}$, with respect to the combined preorder $\leq_{x \cap y}$ defined by:*

$$(a|b) \leq_{x \cap y} (c|d) \quad \text{if and only if} \quad (a|b) \leq_x (c|d) \text{ and } (a|b) \leq_y (c|d).$$

That $\leq_{x \cap y}$ is a deductive relation is straightforward, and the proof of the theorem is also quite straightforward.

In view of the hierarchical relationships between the various deductive relations presented here, the above theorem significantly simplifies matters since:

(a) All non-trivial deductive relations in the hierarchy can be built up by combining the four elementary deductive relations.

(b) The deductively closed sets with respect to a combined deductive relation include all the intersections of the deductively closed sets with respect to their constituent deductive relations.

(c) For any initial subset J of conditionals, to determine the deductively closed sets generated by J with respect to the various combined deductive relations in the hierarchy, start with the intersections of the deductively closed sets with respect to the constituent deductive relations. All such intersections will be DCSs with respect to the combined relation, but not all DCSs with respect to a combined deductive relation are intersections of

DCSs of more elementary deductive relations, although those generated by a single conditional will be shown to be.

3.3.2. Conjunction of conditionals implies disjunction. Note also that for any deductive relation $\leq_x$ with the Boolean extension property, $(a|b) \wedge (c|d) \leq_x (a|b) \vee (c|d)$ always holds. It then follows from either of the defining properties of a deductive relation that for any deductively closed set H

$$\text{If } (a|b) \in H \text{ and } (c|d) \in H \text{ then } (a|b) \vee (c|d) \in H.$$

But it is not in general true (unless $\leq_\vee$ is also assumed) that

$$\text{If } (a|b) \in H \text{ then for all } (c|d), (a|b) \vee (c|d) \in H.$$

3.4. Generators of a deductively closed set of conditionals

In practice we are interested in determining what conditionals can be deduced, and in what sense deduced, from a given set J of conditionals.

3.4.1. Definition (deductive extension). If J is any subset of $\mathscr{B}/\mathscr{B}$, let $H_x(J)$ denote the smallest deductively closed subset with respect to $\leq_x$ that includes J. We say that $H_x(J)$ is the deductive extension of J with respect to $\leq_x$, or that J generates $H_x(J)$ with respect to $\leq_x$, or that J "$\leq_x$-implies" the DCS $H_x(J)$. A DCS is principal if it is generated by a single conditional.

3.4.2. Theorem on principal deductively closed sets. With respect to any deductive relation $\leq_x$ among the 13 in the hierarchy, the deductively closed set generated by a single conditional $(a|b)$ is the set of conditionals that subsume it with respect to the deductive relation. That is, with respect to any preorder $\leq_x$ among the 13 in the hierarchy the principal DCS generated by an individual conditional $(a|b)$ is the set $H_x\{(a|b)\} = \{(y|z) : (a|b) \leq_x (y|z)\}$ of all conditionals that are implied by $(a|b)$ with respect to $\leq_x$. $H_x\{(a|b)\}$ will be denoted by $H_x(a|b)$.

Proof of Theorem 3.4.2. The proof follows from the fact that for $x \in \{\text{ap}, \text{tr}, \text{nf}, \text{ip}\}$, $H_x(a|b)$ is closed with respect to $\leq_x$, by some easy algebraic steps and so too for the other deductive relations above them in the hierarchy since they consist of combinations of the defining properties of the elementary deductive relations. (For $x = 0$, $H_x(a|b)$ is trivially closed with respect to $\leq_x$.) Thus conjunctive closure (of Definition 3.1.5) holds for $H_x(a|b)$ in these cases. Deductive closure (of Definition 3.1.5) also holds for $H_x(a|b)$ by the transitivity property of all deductive relations $\leq_x$. Thus for all deductive relations in the hierarchy, $H_x(a|b)$ is deductively closed and includes $(a|b)$. Clearly, every deductively closed set containing $(a|b)$ must also include $H_x(a|b)$. So $H_x(a|b)$ is the deductively closed set generated by $\{(a|b)\}$. □

With respect to each of the deductive relations in the hierarchy of Fig. 1 it is now possible to completely describe the principal deductively closed sets of conditionals generated by a single conditional (a|b).

3.4.2.1. Principal deductively closed sets of the elementary deductive relations $\leq_{ap}, \leq_{tr}, \leq_{nf}$, and $\leq_{ip}$. For deductive relation $\leq_{ap}$, the DCS generated by the single conditional (a|b) is $H_{ap}(a|b) = \{(c|d) : (a|b) \leq_{ap} (c|d)\} = \{(c|d) : b \leq d\}$, and so

$$H_{ap}(a|b) = \{(x|b \vee y) : \text{any } x, y \text{ in } \mathscr{B}\}.$$

For deductive relation $\leq_{tr}$, the DCS generated by the single conditional (a|b) is

$$\begin{aligned} H_{tr}(a|b) &= \{(c|d) : (a|b) \leq_{tr} (c|d)\} \\ &= \{(c|d) : ab \leq cd\} \\ &= \{(c|d) : ab \leq c, ab \leq d\}, \end{aligned}$$

and so

$$H_{tr}(a|b) = \{(ab \vee x|ab \vee y) : \text{any } x, y \text{ in } \mathscr{B}\}.$$

For deductive relation $\leq_{nf}$, the DCS generated by the single conditional (a|b) is

$$\begin{aligned} H_{nf}(a|b) &= \{(c|d) : (a|b) \leq_{nf} (c|d)\} \\ &= \{(c|d) : a \vee b' \leq c \vee d'\} \\ &= \{(c|d) : d(a \vee b') \leq c\} \\ &= \{((d(a \vee b') \vee x)|d) : \text{any } x, d \text{ in } \mathscr{B}\} \\ &= \{((y(a \vee b') \vee x)|y) : \text{any } x, y \text{ in } \mathscr{B}\}, \end{aligned}$$

and so

$$H_{nf}(a|b) = \{(ab \vee b' \vee x|y) : \text{any } x, y \text{ in } \mathscr{B}\}.$$

For deductive relation $\leq_{ip}$, the DCS generated by the single conditional (a|b) is

$$H_{ip}(a|b) = \{(c|d) : (a|b) \leq_{ip} (c|d)\} = \{(c|d) : b' \leq d'\} = \{(c|d) : d \leq b\},$$

and so

$$H_{ip}(a|b) = \{(x|by) : \text{any } x, y \text{ in } \mathscr{B}\}.$$

These principal DCSs for the elementary deductive relations are given in non-reduced form but the reduced forms can easily be determined.

3.4.2.2. Principal deductively closed sets of non-elementary deductive relations. Using the Conjunction Theorem 3.3.1 for deductive relations, and Theorem 3.4.2 on principal DCSs all the DCSs generated by a single conditional (a|b) by the non-elementary deductive relations can be determined as the intersection of the principal DCSs of the elementary deductive relations. This follows because for a single conditional (a|b),

$$\begin{aligned} H_{x\cap y}(\mathrm{a}|\mathrm{b}) &= \{(\mathrm{c}|\mathrm{d}) : (\mathrm{a}|\mathrm{b}) \leq_{x\cap y} (\mathrm{c}|\mathrm{d})\} \\ &= \{(\mathrm{c}|\mathrm{d}) : (\mathrm{a}|\mathrm{b}) \leq_x (\mathrm{c}|\mathrm{d}) \text{ and } (\mathrm{a}|\mathrm{b}) \leq_y (\mathrm{c}|\mathrm{d})\} \\ &= H_x(\mathrm{a}|\mathrm{b}) \cap H_y(\mathrm{a}|\mathrm{b}). \end{aligned}$$

For deductive relation $\leq_\vee$, the DCS generated by the single conditional $(\mathrm{a}|\mathrm{b})$ is $H_\vee(\mathrm{a}|\mathrm{b}) = H_{\mathrm{ap}\cap\mathrm{tr}}(\mathrm{a}|\mathrm{b}) = H_{\mathrm{ap}}(\mathrm{a}|\mathrm{b}) \cap H_{\mathrm{tr}}(\mathrm{a}|\mathrm{b})$. So by combining constraints expressed above for $H_{\mathrm{ap}}(\mathrm{a}|\mathrm{b})$ and $H_{\mathrm{tr}}(\mathrm{a}|\mathrm{b})$,

$$H_\vee(\mathrm{a}|\mathrm{b}) = \{(\mathrm{ab} \vee x|\mathrm{b} \vee y) : \text{any } x, y \text{ in } \mathscr{B}\}.$$

For deductive relation $\leq_{\mathrm{pm}}$, the DCS generated by the single conditional $(\mathrm{a}|\mathrm{b})$ is $H_{\mathrm{pm}}(\mathrm{a}|\mathrm{b}) = H_{\mathrm{tr}\cap\mathrm{nf}}(\mathrm{a}|\mathrm{b}) = H_{\mathrm{tr}}(\mathrm{a}|\mathrm{b}) \cap H_{\mathrm{nf}}(\mathrm{a}|\mathrm{b})$. So by combining constraints expressed above for $H_{\mathrm{tr}}(\mathrm{a}|\mathrm{b})$ and $H_{\mathrm{nf}}(\mathrm{a}|\mathrm{b})$,

$$H_{\mathrm{pm}}(\mathrm{a}|\mathrm{b}) = \{(\mathrm{ab} \vee \mathrm{b}' \vee x|\mathrm{ab} \vee y) : \text{any } x, y \text{ in } \mathscr{B}\}.$$

For deductive relation $\leq_\wedge$, the DCS generated by the single conditional $(\mathrm{a}|\mathrm{b})$ is $H_\wedge(\mathrm{a}|\mathrm{b}) = H_{\mathrm{nf}\cap\mathrm{ip}}(\mathrm{a}|\mathrm{b}) = H_{\mathrm{nf}}(\mathrm{a}|\mathrm{b}) \cap H_{\mathrm{ip}}(\mathrm{a}|\mathrm{b})$. So by combining constraints expressed above for $H_{\mathrm{nf}}(\mathrm{a}|\mathrm{b})$ and $H_{\mathrm{ip}}(\mathrm{a}|\mathrm{b}), H_\wedge(\mathrm{a}|\mathrm{b}) = \{((\mathrm{ab} \vee \mathrm{b}' \vee x)|\mathrm{b}y) : \text{any } x, y \text{ in } \mathscr{B}\}$, which can be simplified as

$$H_\wedge(\mathrm{a}|\mathrm{b}) = \{(\mathrm{ab} \vee x|\mathrm{b}y) : \text{any } x, y \text{ in } \mathscr{B}\}.$$

For deductive relation $\leq_{\mathrm{ec}}$, the DCS generated by the single conditional $(\mathrm{a}|\mathrm{b})$ is simply $\{(x|y) : y = \mathrm{b}\}$. So

$$H_{\mathrm{ec}}(\mathrm{a}|\mathrm{b}) = \{(x|\mathrm{b}) : \text{any } x \text{ in } \mathscr{B}\}.$$

For deductive relation $\leq_{\mathrm{m}\vee}$, the DCS generated by the single conditional $(\mathrm{a}|\mathrm{b})$ is $H_{\mathrm{m}\vee}(\mathrm{a}|\mathrm{b}) = H_{\mathrm{ap}\cap\mathrm{pm}}(\mathrm{a}|\mathrm{b}) = H_{\mathrm{ap}}(\mathrm{a}|\mathrm{b}) \cap H_{\mathrm{pm}}(\mathrm{a}|\mathrm{b})$. So by combining constraints expressed above for $H_{\mathrm{ap}}(\mathrm{a}|\mathrm{b})$ and $H_{\mathrm{pm}}(\mathrm{a}|\mathrm{b})$,

$$H_{\mathrm{m}\vee}(\mathrm{a}|\mathrm{b}) = \{((\mathrm{ab} \vee \mathrm{b}' \vee x)|(\mathrm{b} \vee y)) : \text{any } x, y \text{ in } \mathscr{B}\}.$$

(Note that $H_{\mathrm{m}\vee}(\mathrm{a}|\mathrm{b})$ can also be expressed as $H_{\mathrm{ap}\cap\mathrm{nf}}(\mathrm{a}|\mathrm{b})$ or as $H_{\vee\cap\mathrm{nf}}(\mathrm{a}|\mathrm{b})$ or as $H_{\vee\cap\mathrm{pm}}(\mathrm{a}|\mathrm{b})$ with equivalent results.)

For deductive relation $\leq_{\mathrm{m}\wedge}$, the DCS generated by the single conditional $(\mathrm{a}|\mathrm{b})$ is $H_{\mathrm{m}\wedge}(\mathrm{a}|\mathrm{b}) = H_{\mathrm{tr}\cap\mathrm{ip}}(\mathrm{a}|\mathrm{b}) = H_{\mathrm{tr}}(\mathrm{a}|\mathrm{b}) \cap H_{\mathrm{ip}}(\mathrm{a}|\mathrm{b})$. So by combining constraints expressed above for $H_{\mathrm{tr}}(\mathrm{a}|\mathrm{b})$ and $H_{\mathrm{ip}}(\mathrm{a}|\mathrm{b}), H_{\mathrm{m}\wedge}(\mathrm{a}|\mathrm{b}) = \{((\mathrm{ab} \vee x)|(\mathrm{ab} \vee y\mathrm{b})) : \text{any } x, y \text{ in } \mathscr{B}\}$. So,

$$H_{\mathrm{m}\wedge}(\mathrm{a}|\mathrm{b}) = \{(\mathrm{a} \vee x|\mathrm{ab} \vee y\mathrm{b}) : \text{any } x, y \text{ in } \mathscr{B}\}.$$

(Note again that $H_{\mathrm{m}\wedge}(\mathrm{a}|\mathrm{b})$ can also be expressed as $H_{\mathrm{tr}\cap\wedge}(\mathrm{a}|\mathrm{b})$ or as $H_{\mathrm{pm}\cap\mathrm{ip}}(\mathrm{a}|\mathrm{b})$ or as $H_{\mathrm{pm}\cap\wedge}(\mathrm{a}|\mathrm{b})$ with equivalent results.)

For the deductive relation $\leq_{\mathrm{bo}}$ the DCS generated by the single conditional $(\mathrm{a}|\mathrm{b})$ is

$$H_{\mathrm{bo}}(\mathrm{a}|\mathrm{b}) = \{((\mathrm{a} \vee x)|\mathrm{b}) : \text{any } x \text{ in } \mathscr{B}\}.$$

Having described the principal DCSs of the 11 non-trivial deductive relations in the hierarchy, they can be used to build up the DCSs generated by more than a single conditional. In this regard, Dubois and Prade [18, Definition 1, p. 1719], adapt a theorem of Adams [4, Theorem 1, p. 52] expressed in a probability context to define the logical entailment of a set of conditionals J. Their construct is similar to what below is called the conjunctive closure:

3.4.3. Deductive extension theorem. Let J be a subset of $\mathscr{B}/\mathscr{B}$ and define the conjunctive closure $D(J)$ of J by:

$$D(J) = \left\{ \bigwedge_{i=1}^{n} (\mathrm{a}|\mathrm{b})_i,\ n \text{ finite},\ (\mathrm{a}|\mathrm{b})_i \in J \right\}.$$

That is, $D(J)$ is the set of all finite conjunctions of any members of J. Then for any deductive relation in the hierarchy except for $x \in \{\mathrm{tr}, \vee\}$, the set $H_x(J)$ defined by

$$H_x(J) = \{(\mathrm{c}|\mathrm{d}) : (\mathrm{a}|\mathrm{b}) \leq_x (\mathrm{c}|\mathrm{d}), (\mathrm{a}|\mathrm{b}) \in D(J)\}$$

that is, the set of all conditionals implied with respect to $\leq_x$ by some member of $D(J)$, is the deductively closed set with respect to $\leq_x$ generated by J. In symbols,

$$H_x(J) = \left\{ (\mathrm{c}|\mathrm{d}) : \exists (\mathrm{a}|\mathrm{b})_i \in J,\ i = 1 \text{ to } n \text{ (finite) such that } \bigwedge_{i=1}^{n} (\mathrm{a}|\mathrm{b})_i \leq_x (\mathrm{c}|\mathrm{d}) \right\}.$$

Corollary of deductive extension theorem. Since $H_x(J)$ is the union of the all conditionals that are implied by some individual member of $D(J)$, it follows by Theorem 3.4.2 on principal deductively closed sets that for all deductive relations $\leq_x$ in the hierarchy except for $\leq_{\mathrm{tr}}$ and $\leq_\vee$, the DCS generated by a set of conditionals J is the union of the principal DCSs generated by the individual members of the set $D(J)$ of all finite conjunctions of the conditionals in J.

Proof of Theorem 3.4.3. First property (2) of Definition 3.1.5: If $(\mathrm{a}|\mathrm{b}) \in H_x(J)$ and $(\mathrm{a}|\mathrm{b}) \leq_x (\mathrm{c}|\mathrm{d})$, then there exists $(\mathrm{q}|\mathrm{p}) \in D(J)$ such that $(\mathrm{q}|\mathrm{p}) \leq_x (\mathrm{a}|\mathrm{b})$. So by transitivity of $\leq_x, (\mathrm{q}|\mathrm{p}) \leq_x (\mathrm{c}|\mathrm{d})$. Therefore, $(\mathrm{c}|\mathrm{d}) \in H_x(J)$. That shows property (2) for any deductive relation $\leq_x$. To show property (1), suppose that $(\mathrm{a}|\mathrm{b}) \in H_x(J)$ and $(\mathrm{c}|\mathrm{d}) \in H_x(J)$. So there exist $(\mathrm{q}|\mathrm{p}) \in D(J)$ and $(\mathrm{s}|\mathrm{r}) \in D(J)$ such that $(\mathrm{q}|\mathrm{p}) \leq_x (\mathrm{a}|\mathrm{b})$ and $(\mathrm{s}|\mathrm{r}) \leq_x (\mathrm{c}|\mathrm{d})$. Now, $(\mathrm{q}|\mathrm{p}) \wedge (\mathrm{s}|\mathrm{r}) \in D(J)$ because the conjunction of two finite conjunctions of elements of J is a finite conjunction of elements of J. It follows (by the next lemma) that except for $x \in \{\mathrm{tr}, \vee\}, (\mathrm{q}|\mathrm{p}) \wedge (\mathrm{s}|\mathrm{r}) \leq_x (\mathrm{a}|\mathrm{b}) \wedge (\mathrm{c}|\mathrm{d})$. Therefore $(\mathrm{a}|\mathrm{b}) \wedge (\mathrm{c}|\mathrm{d}) \in H_x(J)$. So property (1) holds. Finally, $H_x(J)$ is the smallest deductively closed subset with respect to $\leq_x$ that includes J because $H_x(J)$ includes $D(J)$, and $D(J)$ includes J. So $H_x(J)$ includes J. Secondly, any deductively closed set that includes J, by repeated application of property (1), must include all finite conjunctions of elements of J, and so, by property (2), must include any (c|d) for which there is an $(a|b) \in J$ with $(a|b) \leq_x (c|d)$. Concerning the deductive relations $\leq_x$ for $x \in \{tr, \vee\}$, a counter example to the theorem is provided by the set $J = \{(1|b), (1|b')\}$ in the nine element conditional algebra $\mathscr{B}/\mathscr{B}$ generated by Boolean algebra $\mathscr{B} = \{1, b, b', 0\}$. (See the four DCSs of $\leq_{tr}$ in Table 1.) For this set J, $D(J) = \{(1|b), (1|b'), 1\}$, and so by the theorem $H_{tr}(J)$ would be $\{(e|f) : (a|b) \leq_{tr} (e|f), (a|b) \in D(J)\} = \{1, (1|b), (1|b'), b, b'\}$. However, this latter set is not a DCS with respect to $\leq_{tr}$ because it obviously also generates 0 from b and b′. The same counter example also works for $\leq_{\vee}$. See Table 3, DCSs #4, #5 and #11. Here again, as for $\leq_{tr}$, the union of the principal DCSs generated by the conditionals in $D(J) = \{(1|b), (1|b'), 1\}$ is not a DCS. □

3.4.4. Lemma for deductive extension theorem. Let $\leq_x$ be any deductive relation in the hierarchy other than $\leq_{tr}$ or $\leq_{\vee}$. If $(q|p) \leq_x (a|b)$ and $(s|r) \leq_x (c|d)$, then $(q|p) \wedge (s|r) \leq_x (a|b) \wedge (c|d)$. For $x \in \{tr, \vee\}$, this is not necessarily true.

Table 1
Deductively closed sets of $\mathscr{B}/\mathscr{B} = \{0, b, b', 1, (0|b), (1|b), (0|b'), (1|b'), U\}$ with respect to the pre-orders (deductive relations) $\leq_{ap}, \leq_{tr}, \leq_{nf}$, and $\leq_{ip}$

Preorders	Nine conditionals of $\mathscr{B} = \{0, b, b', 1\}$									
	DCS #	1	b	b′	(1|b)	(1|b′)	0	(0|b)	(0|b′)	U
$\leq_{ap}$	1	G	G	G			G			
	2	+	+	+	G		+	G		
	3	+	+	+		G	+		G	
	4	+	+	+	+	+	+	+	+	G
	5	+	+	+	G	H	+	G	H	
$\leq_{tr}$	1	+	+	+	+	+	G	G	G	G
	2	G								
	3	+	G		G					
	4	+		G		G				
$\leq_{nf}$	1	+	+	+	+	+	G	+	+	+
	2	G			G	G				G
	3	+	G		+	+			G	+
	4	+		G	+	+		G		+
$\leq_{ip}$	1	G	G	G	+	+	G	+	+	+
	2				G			G		+
	3					G			G	+
	4									G

Key. # – Numbered deductively closed set (DCS) for a preorder; + – Included in the DCS of that row; G, H – Generators of the DCS of that row; one of each present in a row is required to generate that row.

Proof of Lemma 3.4.4. First a counter-example for $x \in \{\mathrm{tr}, \vee\}$: Consider that $(0|0) \leq_\vee (0|1)$ and $(1|1) \leq_\vee (1|1)$. Now $(0|0) \wedge (1|1) = (1|1)$ and $(0|1) \wedge (1|1) = (0|1)$. But $(1|1) \leq_{\mathrm{tr}} (0|1)$ is always false and so too is the stronger $(1|1) \leq_\vee (0|1)$. Concerning $x = \mathrm{ap}$, suppose $(\mathrm{q}|\mathrm{p}) \leq_{\mathrm{ap}} (\mathrm{a}|\mathrm{b})$ and $(\mathrm{s}|\mathrm{r}) \leq_{\mathrm{ap}} (\mathrm{c}|\mathrm{d})$. So $\mathrm{p} \leq \mathrm{b}$ and $\mathrm{r} \leq \mathrm{d}$, and so $\mathrm{p} \vee \mathrm{r} \leq \mathrm{b} \vee \mathrm{d}$. Therefore

$$\begin{aligned}(\mathrm{q}|\mathrm{p}) \wedge (\mathrm{s}|\mathrm{r}) &= (\mathrm{qpr}' \vee \mathrm{p}'\mathrm{sr} \vee \mathrm{qpsr})|(\mathrm{p} \vee \mathrm{r}) \\ &\leq_{\mathrm{ap}} (\mathrm{abd}' \vee \mathrm{b}'\mathrm{cd} \vee \mathrm{abcd})|(\mathrm{b} \vee \mathrm{d}) \\ &= (\mathrm{a}|\mathrm{b}) \wedge (\mathrm{c}|\mathrm{d}).\end{aligned}$$

A similar argument with the inequalities reversed works for $x = \mathrm{ip}$. Concerning $x = \mathrm{nf}$, suppose that both $(\mathrm{q}|\mathrm{p}) \leq_{\mathrm{nf}} (\mathrm{a}|\mathrm{b})$ and $(\mathrm{s}|\mathrm{r}) \leq_{\mathrm{nf}} (\mathrm{c}|\mathrm{d})$. Then

$$\begin{aligned}(\mathrm{q}|\mathrm{p}) \wedge (\mathrm{s}|\mathrm{r}) &= (\mathrm{qpr}' \vee \mathrm{p}'\mathrm{sr} \vee \mathrm{qpsr})|(\mathrm{p} \vee \mathrm{r}) \\ &\leq_{\mathrm{nf}} (\mathrm{abd}' \vee \mathrm{b}'\mathrm{cd} \vee \mathrm{abcd})|(\mathrm{b} \vee \mathrm{d}) \\ &= (\mathrm{a}|\mathrm{b}) \wedge (\mathrm{c}|\mathrm{d})\end{aligned}$$

since

$$\begin{aligned}(\mathrm{qpr}' \vee \mathrm{p}'\mathrm{sr} \vee \mathrm{qpsr}) \vee (\mathrm{p} \vee \mathrm{r})' &= (\mathrm{qp} \vee \mathrm{p}')(\mathrm{sr} \vee \mathrm{r}') \\ &= (\mathrm{q} \vee \mathrm{p}')(\mathrm{s} \vee \mathrm{r}') \leq (\mathrm{a} \vee \mathrm{b}')(\mathrm{c} \vee \mathrm{d}') \\ &= (\mathrm{abd}' \vee \mathrm{b}'\mathrm{cd} \vee \mathrm{abcd}) \vee (\mathrm{b} \vee \mathrm{d})'.\end{aligned}$$

That shows the result for $x = \mathrm{nf}$. Since the result holds for $x = \mathrm{nf}$ and for $x = \mathrm{ip}$, the result holds for $x = \wedge$, which just combines these two elementary deductive relations. Now suppose the hypothesis holds for $x = \mathrm{pm}$. So the four inequalities $(\mathrm{q} \vee \mathrm{p}') \leq (\mathrm{a} \vee \mathrm{b}'), \mathrm{qp} \leq \mathrm{ab}, (\mathrm{s} \vee \mathrm{r}') \leq (\mathrm{c} \vee \mathrm{d}')$ and $\mathrm{sr} \leq \mathrm{cd}$ hold. Again, the result holds for $x = \mathrm{nf}$. Therefore, by the first part of the proof, the result holds for $x = \mathrm{pm}$ providing $(\mathrm{qp})(\mathrm{s} \vee \mathrm{r}') \vee (\mathrm{q} \vee \mathrm{p}')(\mathrm{sr}) \leq (\mathrm{ab})(\mathrm{c} \vee \mathrm{d}') \vee (\mathrm{cd})(\mathrm{a} \vee \mathrm{b}')$, and the latter easily follows from these four inequalities. The deductive relation $\leq_{\mathrm{m}\vee}$ is made up of the two elementary deductive relation $\leq_{\mathrm{ap}}$ and $\leq_{\mathrm{nf}}$, for which the result holds. Therefore the result holds for $\leq_{\mathrm{m}\vee}$. The deductive relation $\leq_{\mathrm{m}\wedge}$ is just the conjunction of the two deductive relations $\leq_{\mathrm{pm}}$ and $\leq_\wedge$ for which the result holds. So it holds too for $\leq_{\mathrm{m}\wedge}$. Similar statements can be made for $x = \mathrm{ce}$ and bo, or proved directly. That completes the proof of the lemma. □

It may seem at first disappointing that it is necessary to take every possible finite conjunction of the members of a given generating set J of conditionals in order to determine the deductive consequences of J with respect to most deductive relations. However this is required because the different conditions of the members of J need to be disjoined ($\vee$) in all possible ways as they are when conditionals are conjoined or disjoined according to the operations. A single conditional, say the conjunction of all members of a finite set J, will have the largest possible condition, the disjunction of all conditions appearing in the member of J, and so will generally miss the smaller conditions of the individual member of J. However, as proved in the next theorem, for three of the deductive relations this single conditional by itself generates the same DCS as J.

3.4.5. Theorem on deductively closed sets of $\leq_{\mathrm{nf}}, \leq_{\mathrm{ip}}$ and $\leq_{\wedge}$. The DCS with respect to $\leq_{\mathrm{nf}}, \leq_{\mathrm{ip}}$ or $\leq_{\wedge}$ generated by a finite set J is principal, and is generated by the conditional proposition $(\alpha|\beta)_J$ defined by:

$$(\alpha|\beta)_J = \bigwedge_{(\mathrm{a}|\mathrm{b})_i \in J} (\mathrm{a}|\mathrm{b})_i$$

This is not true for the other deductive relations in the hierarchy, and in fact, their finite DCSs are not in general principal.

Proof of Theorem 3.4.5. Since J is a finite set of conditionals $\{(\mathrm{a}|\mathrm{b})_i\}$, it follows by repeated application of property (1) of Definition 3.1.5 that for all deductive relations $\leq_x$, $(\mathrm{a}|\mathrm{b})_J$ is a conditional proposition in the DCS with respect to $\leq_x$ generated by J. Furthermore, for x in $\{\mathrm{nf}, \mathrm{ip}, \wedge\}$, $(\alpha|\beta)_J \leq_x (\mathrm{a}|\mathrm{b})_i$, for all $(\mathrm{a}|\mathrm{b})_i$. [For $x = \mathrm{nf}$, this follows from the fact that $(\mathrm{a}|\mathrm{b}) \wedge (\mathrm{c}|\mathrm{d}) = [(\mathrm{a} \vee \mathrm{b}')(\mathrm{c} \vee \mathrm{d}')|(\mathrm{b} \vee \mathrm{d})]$, which is not false on $(\mathrm{a} \vee \mathrm{b}')(\mathrm{c} \vee \mathrm{d}') \vee (\mathrm{b} \vee \mathrm{d})' = (\mathrm{a} \vee \mathrm{b}')(\mathrm{c} \vee \mathrm{d}')$, and $(\mathrm{a} \vee \mathrm{b}')(\mathrm{c} \vee \mathrm{d}') \leq (\mathrm{a} \vee \mathrm{b}')$. So$(\mathrm{a}|\mathrm{b}) \wedge (\mathrm{c}|\mathrm{d}) \leq_{\mathrm{nf}} (\mathrm{a}|\mathrm{b})$. Repeated applieation then yields the result for $x = \mathrm{nf}$. For $x = \mathrm{ip}$, this result follows because eonjunction of conditionals always yields a resulting eonditional whose condition includes the eonditions of all the components of the conjunction, and therefore this conjunction will imply each of the component conditionals with respect to $x = \mathrm{ip}$. For $x = \wedge$, by definition, the conjunction of eonditionals implies with respect to $\leq_{\wedge}$ the components of the conjunction. Alternately, since the defining properties of $\leq_{\wedge}$ consist of the combined eharacteristics of $\leq_{\mathrm{ip}}$ and $\leq_{\mathrm{nf}}$, it follows that $\leq_{\wedge}$ too has the property.] So for $x = \mathrm{ip}$, nf or $\wedge$, the principal DCS with respect to $\leq_x$ generated by $(\alpha|\beta)_J$ includes all the conditional propositions of J. So with respect to $\leq_x$, $(\alpha|\beta)_J$ generates all the conditional propositions of J and is generated by J. For the other deductive relations in the hierarchy, Section 4.3 will provide examples of non-principal DCSs, which therefore cannot be generated by any single conditional including $(\alpha|\beta)_J$. That completes the proof of Theorem 3.4.5. □

When a new conditional is adjoined to a collection of conditionals, or if two sets of conditionals are combined, the resulting collection has new deductions.

3.4.6. Theorem on additional deductive information. For all deductive relations $\leq_x$ in the hierarchy except for $x \in \{\mathrm{tr}, \vee\}$, the deductively closed set generated by a DCS J and an additional conditional proposition $(\mathrm{c}|\mathrm{d})$ is the set of all conditionals implied with respect to $\leq_x$ by the conjunction of some conditional in J with the conditional $(\mathrm{c}|\mathrm{d})$. That is,

$$H_x(J \cup \{(c|d)\}) = \{(q|p) : \exists(a|b) \in J \text{ such that } (a|b) \wedge (c|d) \leq_x (q|p)\}$$

and more generally, if K is another DCS of conditional propositions with respect to $\leq_x, x \notin \{tr, \vee\}$ and in the hierarchy, then the DCS generated by the union of J and K is

$$H_x(J \cup K) = \{(q|p) : \exists(a|b) \in J \text{ and } (c|d) \in K \text{ such that } (a|b) \wedge (c|d) \leq_x (q|p)\}$$

which is the set of conditionals implied with respect to $\leq_x$ by the conjunction of some conditional in J with some conditional from K.

Proof of Theorem 3.4.6. $H_x(J \cup K)$ is a DCS since if $(q|p) \in H_x(J \cup K)$ and $(s|r) \in H_x(J \cup K)$ then there are $(a|b) \in J$ and $(c|d) \in K$ with $(a|b) \wedge (c|d) \leq_x (q|p)$. Similarly there exist $(e|f) \in J$ and $(g|h) \in K$ with $(e|f) \wedge (g|h) \leq_x (s|r)$. So by the Lemma of the Extension Theorem $(a|b) \wedge (c|d) \wedge (e|f) \wedge (g|h) \leq_x (q|p) \wedge (s|r)$. By the commutative law for conditionals this can be expressed as $(a|b) \wedge (e|f) \wedge (c|d) \wedge (g|h) \leq_x (q|p) \wedge (s|r)$, with $(a|b) \wedge (e|f) \in J$ and $(c|d) \wedge (g|h) \in K$. So $(q|p) \wedge (s|r) \in H_x(J \cup K)$. That shows that $H_x(J \cup K)$ is closed under conjunction. Now suppose $(q|p) \in H_x(J \cup K)$ and that $(q|p) \leq_x (s|r)$. So there are $(a|b) \in J$ and $(c|d) \in K$ with $(a|b) \wedge (c|d) \leq_x (q|p)$. By transitivity it easily follows that $(a|b) \wedge (c|d) \leq_x (s|r)$. So $(s|r) \in H_x(J \cup K)$. That completes the proof. □

By defining as usual $(J \wedge K)$ to be $\{(a|b) \wedge (c|d) : (a|b) \in J, (c|d) \in K\}$, it follows that for all deductive relations $\leq_x$ in the hierarchy except $x \in \{tr, \vee\}$, $(J \cup K) \subseteq (J \wedge K) \subseteq H_x(J \cup K)$, for DCSs J and K, but the equalities may not hold. The deductively closed set generated by the union of two DCSs can be something more than the simple conjunction of the conditionals of one DCS with those of the other DCS. This simple conjunction may not be a DCS. Here again, there is a difference between the situation for conditional propositions and the situation for Boolean propositions. In the Boolean case, $J \wedge K = H_x(J \cup K)$ always holds. But even in the Boolean case the conjunction $J \wedge K$ of two DCSs can be larger than the simple union $(J \cup K)$ of the component DCSs.

3.4.7. Non-elementary examples of deductively closed sets. With Theorem 3.4.6 it is easy to specify many non-elementary examples of DCSs with respect to various deductive relations.

For instance, suppose $J = \{(a|b), (b|c)\}$ has two conditionals and we want the DCS with respect to $\leq_\wedge$, generated by J. By Theorem 3.4.5 all DCSs with respect to $\leq_\wedge$ are principal and the generating conditional in this case is the conjunction of the members of J, namely, $(a|b) \wedge (b|c) = (ab|(b \vee c))$. So

$$\begin{aligned} H_\wedge(J) &= H_\wedge(\mathrm{ab}|(\mathrm{b} \vee \mathrm{c})) \\ &= \{((\mathrm{ab})(\mathrm{b} \vee \mathrm{c}) \vee x)|y(\mathrm{b} \vee \mathrm{c}) : \text{any } x, y \text{ in } \mathscr{B}\} \\ &= \{((\mathrm{ab} \vee x)|y(\mathrm{b} \vee \mathrm{c})) : \text{any } x, y \text{ in } \mathscr{B}\} \\ &= \{(\mathrm{ab} \vee x|y\mathrm{b} \vee y\mathrm{c}) : \text{any } x, y \text{ in } \mathscr{B}\}. \end{aligned}$$

But if instead we wish to determine the DCS with respect to $\leq_{\mathrm{pm}}$ generated by $J = \{(\mathrm{a}|\mathrm{b}), (\mathrm{b}|\mathrm{c})\}$, that is the $\leq_{\mathrm{pm}}$-implications of J, then we must first form $\mathrm{D}(J) = \{(\mathrm{a}|\mathrm{b}), (\mathrm{c}|\mathrm{d}), (\mathrm{ab}|(\mathrm{b} \vee \mathrm{c}))\}$ and then take the union of the principal DCSs generated by the members of $D(J)$. So $H_{\mathrm{pm}}(J) = H_{\mathrm{pm}}(\mathrm{a}|\mathrm{b}) \cup H_{\mathrm{pm}}(\mathrm{c}|\mathrm{d}) \cup H_{\mathrm{pm}}(\mathrm{ab}|(\mathrm{b} \vee \mathrm{c}))$. Each of these three DCSs could be expressed as intersections of DCSs with respect to more elementary deductive relations since in general, $H_{x \cap y}(\mathrm{a}|\mathrm{b}) = H_x(\mathrm{a}|\mathrm{b}) \cap H_y(\mathrm{a}|\mathrm{b})$. But more directly, using the results of Section 3.4.2.2 it follows that $H_{\mathrm{pm}}(\mathrm{a}|\mathrm{b}) \subseteq H_{\mathrm{pm}}(\mathrm{ab}|(\mathrm{b} \vee \mathrm{c}))$, and so

$$\begin{aligned} H_{\mathrm{pm}}(J) &= H_{\mathrm{pm}}(\mathrm{c}|\mathrm{d}) \cup H_{\mathrm{pm}}(\mathrm{ab}|(\mathrm{b} \vee \mathrm{c})) \\ &= \{(\mathrm{cd} \vee \mathrm{d}' \vee x|\mathrm{cd} \vee y) : \text{any } x, y \text{ in } \mathscr{B}\} \\ &\quad \cup \{(\mathrm{ab} \vee \mathrm{b}'\mathrm{c}' \vee w|\mathrm{ab} \vee z) : \text{any } w, z \text{ in } \mathscr{B}\}. \end{aligned}$$

Note that since $(\mathrm{a}|\mathrm{b}) \leq_{\mathrm{pm}} (\mathrm{c}|\mathrm{d})$ for any $(\mathrm{c}|\mathrm{d})$ in $H_{\mathrm{pm}}(\mathrm{a}|\mathrm{b})$, the probabilistic monotonicity of $\leq_{\mathrm{pm}}$ means that $P(\mathrm{a}|\mathrm{b})$ is a lower bound on the probabilities of the conditionals in $H_{\mathrm{pm}}(\mathrm{a}|\mathrm{b})$.

3.4.8. Applications of the deductive relations. We are used to making Boolean deductions in essentially one way. We show "A implies B" by showing that event A is a subset of event B, that every instance of A is an instance of B. But conditionals have two components, which complicates things, and results in several different kinds of implication, each good for a different purpose depending upon what properties one wants to imply in a deduced conditional. It may be that one wishes to imply the simple truth of one conditional $(\mathrm{c}|\mathrm{d})$ from another one $(\mathrm{a}|\mathrm{b})$, in which case, the deductive relation $\leq_{\mathrm{tr}}$ would be appropriate. One could then conclude that $P(\mathrm{ab}) \leq P(\mathrm{cd})$. If on the other hand one wishes to deduce the non-falsity of one conditional from another, then $\leq_{\mathrm{nf}}$ would be the appropriate deductive relation. Every instance of $(\mathrm{a}|\mathrm{b})$ being true or inapplicable is an instance of $(\mathrm{c}|\mathrm{d})$ being true or inapplicable, and no matter what, $P(\mathrm{a} \vee \mathrm{b}') \leq P(\mathrm{c} \vee \mathrm{d}')$ for the two conditionals. This implication would be appropriate if one wished just to preserve Boolean non-falsity for logical purposes but was not concerned about the conditional probability of the conditionals when they were partially false. The implication $\leq_{\mathrm{pm}}$ combines the defining characteristics of $\leq_{\mathrm{tr}}$ and $\leq_{\mathrm{nf}}$ so that both characteristics must be true for the implication to hold, in which case it follows that $P(\mathrm{a}|\mathrm{b}) \leq P(\mathrm{c}|\mathrm{d})$. If instead of conditional probability, one wishes to have that $(\mathrm{a}|\mathrm{b}) \wedge (\mathrm{c}|\mathrm{d}) = (\mathrm{a}|\mathrm{b})$ whenever $(\mathrm{a}|\mathrm{b})$ implies $(\mathrm{c}|\mathrm{d})$, as is always true in Boolean algebra, then one would use the deductive relation $\leq_\wedge$. Then one could only say that $P(a \vee b') \leq P(c \vee d')$ and that $P(b') \leq P(d')$. On the other hand, if one wishes to have the property $(a|b) \vee (c|d) = (c|d)$ whenever $(a|b)$ implies $(c|d)$, as is always true in (unconditioned) Boolean algebra, then $\leq_{\vee}$ would be appropriate combining the characteristic of $\leq_{tr}$ with that of $\leq_{ap}$ and then it would also follow that $P(ab) \leq P(cd)$ and $P(b) \leq P(d)$. At the expense of another requirement for the deduction of $(c|d)$ by $(a|b)$, one can have the advantages of $\leq_{pm}$ combined with those of $\leq_{\wedge}$ in $\leq_{m\wedge}$, or the advantages of $\leq_{pm}$ combined with those of $\leq_{\vee}$ in $\leq_{m\vee}$.

3.4.9. The penguin problem. It has become traditional to see how theoretical results can be applied to the following problem: "birds fly", "penguins are birds", and "penguins don't fly". What are the implications of these three statements with respect to the various deductive relations?

Representing these three conditionals as $(F|B)$, $(B|p)$ and $(F'|p)$, respectively, then $J = \{(F|B), (B|p), (F'|p)\}$, and the conjunctive closure

$$\begin{aligned} D(J) &= \{(F|B),(B|p),(F'|p),(F|B)(B|p),(B|p)(F'|p),(F|B)(F'|p),(F|B)(B|p)(F'|p)\} \\ &= \{(F|B),(B|p),(F'|p),(BF|B\vee p),(BF'|p),(BFp'\vee B'F'p|B\vee p),(BFp'|B\vee p)\}. \end{aligned}$$

By Theorem 3.4.5, for $x=$ ip, nf or $\wedge, H_x(J) = H_x((F|B)(B|p)(F'|p)) = H_x(BFp'|B \vee p)$. So $H_{ip}(BFp'|B \vee p) = \{(x|(B \vee p)y)$: any $x, y \in \mathscr{B}\}$. That is, the DCS with respect to $\leq_{ip}$ generated by J is the set of all conditionals whose condition is a subset of $(B \vee p)$. Similarly,

$$\begin{aligned} H_{nf}(J) &= H_{nf}(BFp'|B \vee p) \\ &= \{(BFp' \vee (B \vee p)' \vee x|y) : \text{any } x, y \in \mathscr{B}\} \\ &= \{(F \vee B')p' \vee x|y) : \text{any } x, y \in \mathrm{B}\}, \end{aligned}$$

which is the set of all conditionals whose conclusion is a superset of $(F \vee B')p'$, which is $(F'B \vee p)'$, the negation of a penguin or non-flying bird. Similarly,

$$H_{\wedge}(J) = H_{\wedge}(BFp'|B \vee p) = \{(BFp' \vee x|(B \vee p)y) : \text{any } x, y \in \mathscr{B}\},$$

the set of all conditionals whose condition is a subset of $(B\vee p)$, the birds or penguins, and whose conclusion is a superset of BFp', the non-penguin flying birds.

If by definition every penguin is a bird, that is, if $p \leq B$, then $(B \vee p) = B$, and simplifications yield that

$$\begin{aligned} H_{ip}(J) &= \{(x|By) : \text{any } x, y \in \mathscr{B}\}, \\ H_{nf}(J) &= \{(Fp' \vee B' \vee x|y) : \text{any } x, y \in \mathscr{B}\} \quad \text{and} \\ H_{\wedge}(J) &= \{(BFp' \vee x|By) : \text{any } x, y \in \mathscr{B}\}. \end{aligned}$$

Concerning the other deductive relations, except for $x = \vee$ or $x =$ tr, by the Corollary to Theorem 3.4.3

$$H_x(J) = H_x(\mathrm{F}|\mathrm{B}) \cup H_x(\mathrm{B}|\mathrm{p}) \cup H_x(\mathrm{F}'|\mathrm{p}) \cup H_x(\mathrm{BF}|\mathrm{B} \vee \mathrm{p}) \cup H_x(\mathrm{BF}'|\mathrm{p})$$
$$\cup H_x(\mathrm{BFp}' \vee \mathrm{B}'\mathrm{F}'\mathrm{p}|\mathrm{B} \vee \mathrm{p}) \cup H_x(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}).$$

But some of these include the others:

For all non-trivial deductive relations $\leq_x$ in the hierarchy, $(\mathrm{BF}'|\mathrm{p}) \leq_x (\mathrm{B}|\mathrm{p})$ and $(\mathrm{BF}'|\mathrm{p}) \leq_x (\mathrm{F}'|\mathrm{p})$. (This can also be stated in terms of the Boolean deduction relation $\leq_{\mathrm{bo}}$ at the top of the hierarchy.) So $H_x(\mathrm{BF}'|\mathrm{p}) \supseteq H_x(\mathrm{B}|\mathrm{p}) \cup H_x(\mathrm{F}'|\mathrm{p})$. Furthermore, for all such $\leq_x$, $(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \leq_x (\mathrm{BF}|\mathrm{B} \vee \mathrm{p})$ and $(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \leq_x \mathrm{BFp}' \vee \mathrm{B}'\mathrm{F}'\mathrm{p}|\mathrm{B} \vee \mathrm{p})$. So

$$H_x(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \supseteq H_x(\mathrm{BF}|\mathrm{B} \vee \mathrm{p}) \cup H_x(\mathrm{BFp}' \vee \mathrm{B}'\mathrm{F}'\mathrm{p}|\mathrm{B} \vee \mathrm{p}).$$

Therefore, for all deductive relations $\leq_x$, other than $\leq_{\mathrm{tr}}$ or $\leq_{\vee}$,

$$H_x(J) = H_x(\mathrm{F}|\mathrm{B}) \cup H_x(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \cup H_x(\mathrm{BF}'|\mathrm{p}).$$

Let $x = \mathrm{pm}$. Then $(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \leq_x (\mathrm{F}|\mathrm{B})$ by direct application of the definition of $\leq_{\mathrm{pm}}$ in terms of $\leq_{\mathrm{tr}}$ and $\leq_{\mathrm{nf}}$. So $H_x(\mathrm{F}|\mathrm{B}) \subseteq H_x(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p})$ and therefore

$$\begin{aligned} H_{\mathrm{pm}}(J) &= H_{\mathrm{pm}}(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \cup H_{\mathrm{pm}}(\mathrm{BF}'|\mathrm{p}) \\ &= \{((\mathrm{BF} \vee \mathrm{B}')\mathrm{p}' \vee x|\mathrm{BFp}' \vee y) : \text{any } x, y \in \mathscr{B}\} \\ &\quad \cup \{(\mathrm{BF}'\mathrm{p} \vee \mathrm{p}' \vee x|\mathrm{BF}'\mathrm{p} \vee y) : \text{any } x, y \in \mathscr{B}\}. \end{aligned}$$

If $\mathrm{p} \leq \mathrm{B}$, then

$$\begin{aligned} H_{\mathrm{pm}}(J) &= \{((\mathrm{BFp}' \vee \mathrm{B}' \vee x|\mathrm{BFp}' \vee y) : \text{any } x, y \in \mathscr{B}\} \\ &\quad \cup \{(\mathrm{F}'\mathrm{p} \;\vee \mathrm{p}' \vee x|\mathrm{F}'\mathrm{p} \vee y) : \text{any } x, y \in \mathscr{B}\}. \end{aligned}$$

So $H_{\mathrm{pm}}(J)$ consists of all conditionals whose condition includes the flying non-penguin birds and whose conclusion includes the flying non-penguin plus the non-birds together with all conditionals whose condition includes the non-flying penguins and whose conclusion includes $(\mathrm{Fp})'$, the complement of the flying penguins.

Let $x = \mathrm{m}\wedge$. Since $(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \leq_{\mathrm{ip}} (\mathrm{F}|\mathrm{B})$ and $(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \leq_{\mathrm{pm}} (\mathrm{F}|\mathrm{B})$, therefore $(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \leq_{\mathrm{m}\wedge} (\mathrm{F}|\mathrm{B})$. So

$$\begin{aligned} H_{\mathrm{m}\wedge}(J) &= H_{\mathrm{m}\wedge}(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \cup H_{\mathrm{m}\wedge}(\mathrm{BFp}'|\mathrm{p}) \\ &= \{(\mathrm{BFp}' \vee x||\mathrm{BFp}' \vee (\mathrm{B} \vee \mathrm{p})y) : \text{any } x, y \in \mathscr{B}\} \\ &\quad \cup \{(\mathrm{BF}'\mathrm{p} \vee x|\mathrm{BF}'\mathrm{p} \vee y\mathrm{p}) : \text{any } x, y \in \mathscr{B}\}. \end{aligned}$$

If $\mathrm{p} \leq \mathrm{B}$, then

$$\begin{aligned} H_x(J) &= \{(\mathrm{BFp}' \vee x||\mathrm{BFp}' \vee \;\mathrm{B}y\;) : \text{any } x, y \in \mathscr{B}\} \\ &\quad \cup \{(\mathrm{F}'\mathrm{p} \vee x|\;\mathrm{F}\;'\mathrm{p} \vee y\mathrm{p}) : \text{any } x, y \in \mathscr{B}\}. \end{aligned}$$

So the implications of J with respect to $\leq_{\mathrm{m}\wedge}$ are all conditionals whose condition is between BFp' and B, and whose conclusion includes BFp' together with all conditionals whose condition is between F′p and p, and whose conclusion includes F′p.

Let $x = \mathrm{m}\vee$. In this case there is no immediate further simplification. So

$$H_{\mathrm{m}\wedge}(J) = H_{\mathrm{m}\wedge}(\mathrm{F}|\mathrm{B}) \cup H_{\mathrm{m}\wedge}(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \cup H_{\mathrm{m}\wedge}(\mathrm{BFp}'|\mathrm{p}),$$

and as above these can be solved using the results of Section 3.4.2.2. If $\mathrm{p} \leq \mathrm{B}$, then $(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \leq_{\mathrm{bo}} (\mathrm{F}|\mathrm{B})$ and so also $(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \leq_{\mathrm{m}\wedge} (\mathrm{F}|\mathrm{B})$. So

$$\begin{aligned} H_{\mathrm{m}\wedge}(J) &= H_{\mathrm{m}\wedge}(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \cup H_{\mathrm{m}\wedge}(\mathrm{BFp}'|\mathrm{p}) \\ &= \{(\mathrm{BFp}' \vee \mathrm{B}' \vee x|\mathrm{B} \vee y) : \text{any } x, y \in \mathscr{B}\} \\ &\quad \cup \{(\mathrm{F}'\mathrm{p} \vee \mathrm{p}' \vee x|\mathrm{p} \vee y) : \text{any } x, y \in \mathscr{B}\}. \end{aligned}$$

So $H_{\mathrm{m}\wedge}(J)$ is the set of all conditionals whose condition includes the birds and whose conclusion includes $\mathrm{BFp}' \vee \mathrm{B}'$, the non-birds plus the flying non-penguins, together with all conditionals whose condition includes the penguins and whose conclusion includes $\mathrm{F}'\mathrm{p} \vee \mathrm{p}'$, the complement of the flying penguins.

Let $x = \mathrm{ap}$. Since $(\mathrm{F}|\mathrm{B}) \leq_{\mathrm{ap}} (\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p})$, therefore $H_{\mathrm{ap}}(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \subseteq H_{\mathrm{ap}}(\mathrm{F}|\mathrm{B})$ and so $H_{\mathrm{ap}}(J) = H_{\mathrm{ap}}(\mathrm{F}|\mathrm{B}) \cup H_{\mathrm{ap}}(\mathrm{BF}'|\mathrm{p})$. This is all conditionals whose condition is either a superset of B or of p.

Let $x = \mathrm{bo}$. So again by the results of Section 3.4.2.2,

$$\begin{aligned} H_{\mathrm{bo}}(J) = {} & \{(\mathrm{FB} \ \vee x|\mathrm{B}) : \text{any } x \in \mathscr{B}\} \\ & \cup \{(\mathrm{BFp}' \vee x|\mathrm{B} \vee \mathrm{p}) : \text{any } x \in \mathscr{B}\} \\ & \cup \{(\mathrm{BF}' \vee x|\mathrm{p}) : \text{any } x \in \mathscr{B}\}. \end{aligned}$$

If $\mathrm{p} \leq \mathrm{B}$, then

$$H_{\mathrm{bo}}(J) = \{(\mathrm{BFp}' \vee x|\mathrm{B}) : \text{any } x \in \mathscr{B}\} \cup \{(\mathrm{pF}' \vee x|\mathrm{p}) : \text{any } x \in \mathscr{B}\}.$$

That is, the Boolean implications of J are all conditionals whose condition is B, the birds, and whose conclusion is a superset of the flying non-penguins plus all conditionals whose condition is p, the penguins, and whose conclusion is a superset of the non-flying penguins. In other words, the implications are the conditionals "if a bird then all events that include flying non-penguins" together with "if a penguin then all events that include the non-flyers".

Let $x = \mathrm{tr}$. So

$$H_{\mathrm{tr}}(J) = H_{\mathrm{tr}}\{(\mathrm{F}|\mathrm{B}), (\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}), (\mathrm{BF}'|\mathrm{p})\}.$$

Since $(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \leq_{\mathrm{tr}} (\mathrm{F}|\mathrm{B})$, therefore $H_{\mathrm{tr}}(J) = H_{\mathrm{tr}}\{(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}), (\mathrm{BF}'|\mathrm{p})\}$. Since $H_{\mathrm{tr}}(J)$ must include the principal DCSs generated by $(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p})$ and

$$(\mathrm{BF}'|\mathrm{p}), H_{\mathrm{tr}}(J) \supseteq H_{\mathrm{tr}}(\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}) \cup H_{\mathrm{tr}}(\mathrm{BF}'|\mathrm{p}).$$

So

$$H_{\rm tr}(J) \supseteq \{(\mathrm{BFp}' \vee x|\mathrm{BFp}' \vee y) : \text{any } x, y \in \mathscr{B}\}$$
$$\cup \{(\mathrm{BF'p} \vee x|\mathrm{BF'p} \vee y) : \text{any } x, y \in \mathscr{B}\}.$$

$H_{\rm tr}(J)$ must therefore contain the conjunction of any two conditionals from the latter two sets of conditionals. Setting $x = 0$ and $y = 1$ in both sets, $H_{\rm tr}(J)$ therefore includes the conditional $(\mathrm{BFp}'|1) \wedge (\mathrm{BF'p}|1) = (\mathrm{BFp}' \wedge \mathrm{BF'p}|1) = (0|1) = 0$. But then $H_{\rm tr}(J)$ includes all conditionals because $(0|1) \leq_{\rm tr} (\mathrm{e}|\mathrm{f})$ for any conditional $(\mathrm{e}|\mathrm{f})$. Thus with respect to $\leq_{\rm tr}$, J generates all conditionals and so implies a contradiction. (It can be shown that $H_{\rm tr}\{(\mathrm{a}|\mathrm{b})\}$ always includes the unconditioned event (ab) and so $H_{\rm tr}\{(\mathrm{a}|\mathrm{b}), (\mathrm{c}|\mathrm{d})\}$ always includes the unconditioned event (ab)(cd).)

Let $x = \vee$. So

$$H_\vee(J) = H_\vee\{(F|\mathrm{B}), (\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p}), (\mathrm{BF}'|\mathrm{p})\}.$$

Since $(\mathrm{F}|\mathrm{B}) \wedge (\mathrm{BF}'|\mathrm{p}) = (\mathrm{BFp}'|\mathrm{B} \vee \mathrm{p})$, the latter conditional is in the DCS generated by the other two conditionals. So

$$H_\vee(J) = H_\vee\{(\mathrm{F}|\mathrm{B}), (\mathrm{BF}'|\mathrm{p})\}$$
$$\supseteq \{(\mathrm{BF} \vee x|\mathrm{B} \vee y) : \text{any } x, y \in \mathscr{B}\}$$
$$\cup \{(\mathrm{BF'p} \vee \mathrm{w}|\mathrm{p} \vee \mathrm{z}) : \text{any } w, z \in \mathscr{B}\}.$$

Letting $y = \mathrm{p}$, $z = \mathrm{B}$, and $x = 0 = w$, yields that $H_\vee(J)$ includes both $(\mathrm{BF}|\mathrm{B} \vee \mathrm{p})$ and $(\mathrm{BF'p}|\mathrm{p} \vee \mathrm{B})$ and so also their conjunction $(0|\mathrm{B} \vee \mathrm{p})$. Thus all conditional events whose condition is a superset of $(\mathrm{B} \vee \mathrm{p})$ are included in $H_\vee(J)$ because $(0|\mathrm{B} \vee \mathrm{p}) \leq_\vee (\mathrm{e}|\mathrm{B} \vee \mathrm{p} \vee \mathrm{f})$ for any e and f. So J implies a contradiction with respect to the deductive relation $\leq_\vee$ whenever the conditions include all the birds and penguins, $\mathrm{B} \vee \mathrm{p}$. Yet a explicit characterization of the DCS with respect to $\leq_\vee$ generated by $\{(\mathrm{F}|\mathrm{B}), (\mathrm{BF}'|\mathrm{p})\}$ is still an open question.

4. Elementary examples of finite deductively closed sets of conditionals with respect to deductive relations

The simplest examples of deductively closed sets of conditionals with respect to various deductive relations are based on the simplest Boolean algebras $\mathscr{B}$ having the smallest number of atoms. Here is a start of an exhaustive list of all DCSs of the simplest conditional Boolean algebras $\mathscr{B}/\mathscr{B}$. Though simple, they nevertheless have already provided useful examples and counter examples.

4.1. Generating set $\mathscr{B}$ with zero atoms

Only the zero element, 0, has no atoms. There is one proposition, $\{0\}$ and one conditional $(\{0\}|\{0\})$, abbreviated 0 and $(0|0)$, respectively, the latter also denoted U. (Strictly speaking a Boolean algebra $\mathscr{B}$ must have at least two members.)

4.2. Generating with a Boolean algebra $\mathscr{B}$ with one atom

The 2-element Boolean algebra $\{0,1\}$. Two propositions, $\{\{0\}$, and $\{1\}\}$, usually abbreviated as 0 and 1. Three conditionals: $\mathscr{B}/\mathscr{B} = \{(0|1), (1|1), \text{and } (1|0)\}$, where $(0|0) = (1|0)$. These are usually abbreviated 0, 1, U, with U interpreted as "inapplicable" or "undefined".

4.2.1. Deductive relation $\leq_{ap}$

Since the conditional propositions 0 and 1 are applicability equivalent, that is, since $(0 =_{\rm ap} 1)$, they generate the same deductively closed set, namely $H_{\rm ap}(0) = H_{\rm ap}(1) = \{(x|y) : (0|1) \leq_{\rm ap} (x|y)\} = \{(x|y) : 1 \leq y) = \{0, 1\}$. The conditional U generates its own DCS, $H_{\rm ap}(\mathrm{U})$ since U is applicability equivalent only to itself:

$$H_{\rm ap}(\mathrm{U}) = \{(x|y) : (1|0) \leq_{\rm ap} (x|y)\} = \{(x|y) : 0 \leq y\} = \{0, 1, \mathrm{U}\} = \mathscr{B}/\mathscr{B}.$$

4.2.2. Deductive relation $\leq_{ip}$

Since $(0 =_{\rm ip} 1)$ holds and U is inapplicability equivalent only to itself, preorder $\leq_{\rm ip}$ has the same generators as does preorder $\leq_{\rm ap}$. Two conditional propositions are equivalent in applicability or inapplicability if and only if they have equivalent conditions. But $\leq_{\rm ap}$ and $\leq_{\rm ip}$ generate different DCSs:

$$H_{\rm ip}(1) = \{(x|y) : (1|1) \leq_{\rm ip} (x|y)\} = \{(x|y) : y \leq 1\} = \{0, 1, \mathrm{U}\} = \mathscr{B}/\mathscr{B}.$$
$$H_{\rm ip}(\mathrm{U}) = \{(x|y) : (1|0) \leq_{\rm ip} (x|y)\} = \{(x|y) : y \leq 0\} = \{\mathrm{U}\}.$$

4.2.3. Deductive relation $\leq_{nf}$

Since $(1 =_{\rm nf} \mathrm{U})$, U and 1 generate the same DCS, namely $H_{\rm nf}(1) = H_{\rm nf}(\mathrm{U}) = \{1, \mathrm{U}\}$:

$$\begin{aligned} H_{\rm nf}(0) &= \{(x|y) : (0|1) \leq_{\rm nf} (x|y)\} \\ &= \{(x|y) : 0 \leq x \vee y'\} \\ &= \mathscr{B}/\mathscr{B} \\ &= \{0, 1, \mathrm{U}\}. \end{aligned}$$

4.2.4. Deductive relation $\leq_{tr}$

Since $(0 =_{\rm tr} \mathrm{U})$, 0 and U generate the same DCS, namely

$$H_{\rm tr}(0) = H_{\rm tr}(\mathrm{U}) = \{(x|y) : (0|1) \leq_{\rm tr} (x|y)\} = \{(x|y) : 0 \leq xy\} = \mathscr{B}/\mathscr{B} = \{0, 1, U\},$$
$$H_{\rm tr}(1) = \{(x|y) : (1|1) \leq_{\rm tr} (x|y)\} = \{(x|y) : 1 \leq xy\} = \{1\}.$$

So the atom 1 generates its own singleton DCS.

Thus in summary concerning the elementary preorders, $H_{ap}(0) = H_{ap}(1) = \{0,1\}$ and with respect to $\leq_{ap}$ both 0 and 1 are generators. U generates $\mathscr{B}/\mathscr{B}$. $H_{ip}(U) = \{U\}$ while with respect to $\leq_{ip}$ both 0 and 1 generate $\mathscr{B}/\mathscr{B}$. $H_{nf}(1) = H_{nf}(U) = \{1, U\}$ while 0 generates $\mathscr{B}/\mathscr{B}$. $H_{tr}(1) = \{1\}$ while both 0 and U generate $\mathscr{B}/\mathscr{B}$.

Concerning the combinations of elementary deductions, the Conjunction Theorem 3.3.1 for deductive relations allows us to determine most of the DCSs in the hierarchy by simply taking the intersection of DCSs generated by the individual elementary preorders or generated by the constituent deductive relations. The others were determined by the Deductive Extension Theorem, or in the case of $\leq_{ap}$ and $\leq_{V}$, by brute force.

The DCSs of Conditionals for the 3-element Conditional Event Algebra $\mathscr{B}/\mathscr{B} = \{(0|1), (1|1), (1|0)\} = \{0, 1, U\}$ with respect to the combination deductive relations are as follows:

4.2.5. Deductive relation $\leq_{ec}$

By the Conjunction Theorem for deductive relations of Section 3.3.1,

$$H_{ec}(1) = H_{ap\cap ip}(1) = H_{ap}(1) \cap H_{ip}(1) = \{0,1\} \cap \mathscr{B}/\mathscr{B} = \{0,1\},$$
$$H_{ap\cap ip}(0) = H_{ap}(0) \cap H_{ip}(0) = H_{ap}(1) \cap H_{ip}(1) = \{0,1\},$$
$$H_{ap\cap ip}(U) = H_{ap}(U) \cap H_{ip}(U) = \mathscr{B}/\mathscr{B} \cap \{U\} = \{U\}.$$

So $\leq_{ec}$ generates three deductively closed sets including the whole space $\mathscr{B}/\mathscr{B}$. Furthermore, with respect to $\leq_{ec}$, $\mathscr{B}/\mathscr{B}$ is already a finite, non-principal DCS since it is not generated by a single conditional but instead requires U and either 0 or 1. This is another difference from the finite Boolean situation, where all finite DCSs are principal.

4.2.6. Deductive relation $\leq_{V}$

Using the Conjunction Theorem for deductive relations, $\leq_{V} = \leq_{ap\cap tr}$. So

$$H_{V}(1) = H_{ap}(1) \cap H_{tr}(1) = \{0,1\} \cap \{1\} = \{1\},$$
$$H_{V}(0) = H_{ap}(0) \cap H_{tr}(0) = \{0,1\} \cap \{0,1,U\} = \{0,1\},$$
$$H_{V}(U) = H_{ap}(U) \cap H_{tr}(U) = \{0,1,U\} \cap \{0,1,U\} = \mathscr{B}/\mathscr{B}.$$

4.2.7. Deductive relation $\leq_{\wedge}$

$$H_{\wedge}(1) = H_{nf\cap ip}(1) = H_{nf}(1) \cap H_{ip}(1) = \{1,U\} \cap \{0,1,U\} = \{1,U\},$$
$$H_{\wedge}(U) = H_{nf\cap ip}(U) = H_{nf}(U) \cap H_{ip}(U) = \{1,U\} \cap \{U\} = \{U\},$$
$$H_{nf\cap ip}(0) = H_{nf}(0) \cap H_{ip}(0) = \{0,1,U\} \cap \{0,1,U\} = \{0,1,U\} = \mathscr{B}/\mathscr{B}.$$

4.2.8. Deductive relation $\leq_{\mathrm{pm}}$

$$H_{\mathrm{pm}}(1) = H_{\mathrm{tr}\cap\mathrm{nf}}(1) = H_{\mathrm{tr}}(1) \cap H_{\mathrm{nf}}(1) = \{1\} \cap \{1, U\} = \{1\},$$
$$H_{\mathrm{pm}}(0) = H_{\mathrm{tr}\cap\mathrm{nf}}(0) = H_{\mathrm{tr}}(0) \cap H_{\mathrm{nf}}(0) = \{0, 1, U\} \cap \{0, 1, U\} = \mathrm{B/B},$$
$$H_{\mathrm{pm}}(\mathrm{U}) = H_{\mathrm{tr}\cap\mathrm{nf}}(\mathrm{U}) = H_{\mathrm{tr}}(\mathrm{U}) \cap H_{\mathrm{nf}}(\mathrm{U}) = \{0, 1, U\} \cap \{1, U\} = \{1, U\}.$$

4.2.9. Deductive relation $\leq_{\mathrm{m}\vee}$

$\leq_{\mathrm{m}\vee} = \leq_{\mathrm{pm}\cap\vee}$. So

$$H_{\mathrm{m}\vee}(1) = H_{\mathrm{pm}}(1) \cap H_{\vee}(1) = \{1\} \cap \{1\} = \{1\},$$
$$H_{\mathrm{m}} \vee (0) = H_{\mathrm{pm}}(0) \cap H_{\vee}(0) = \{0, 1, \mathrm{U}\} \cap \{0, 1\} = \{0, 1\},$$
$$H_{\mathrm{m}\vee}(\mathrm{U}) = H_{\mathrm{pm}}(\mathrm{U}) \cap H_{\vee}(\mathrm{U}) = \{1, \mathrm{U}\} \cap \{0, 1, \mathrm{U}\} = \{1, \mathrm{U}\}.$$

Since any single one of its elements does not generate $\mathscr{B}/\mathscr{B}$, $\mathscr{B}/\mathscr{B}$ is a non-prineipal DDS with respeet to $\leq_{\mathrm{m}\vee}$ generated by 0 and U.

4.2.10. Deductive relation $\leq_{\mathrm{m}\wedge}$

$\leq_{\mathrm{m}\wedge} = \leq_{\mathrm{pm}\cap\wedge}$. So

$$H_{\mathrm{m}\wedge}(1) = H_{\mathrm{pm}}(1) \cap H_{\wedge}(1) = \{1\} \cap \{1, \mathrm{U}\} = \{1\},$$
$$H_{\mathrm{m}\wedge}(0) = H_{\mathrm{pm}}(0) \cap H_{\wedge}(0) = \{0, 1, \mathrm{U}\} \cap \{0, 1, \mathrm{U}\} = \{0, 1, \mathrm{U}\},$$
$$H_{\mathrm{m}\wedge}(\mathrm{U}) = H_{\mathrm{pm}}(\mathrm{U}) \cap H_{\wedge}(\mathrm{U}) = \{1, \mathrm{U}\} \cap \{\mathrm{U}\} = \{\mathrm{U}\}.$$

In addition $H_{\mathrm{m}\wedge}\{1, \mathrm{U}\} = H_{\mathrm{pm}}\{1, \mathrm{U}\} \cap H_{\wedge}\{1, \mathrm{U}\} = \{1, \mathrm{U}\} \cap \{1, \mathrm{U}\} = \{1, \mathrm{U}\}$. That is, {1,U}is a non-principal DCS with respeet to $\leq_{\mathrm{m}\wedge}$ since it requires two propositions to generate it.

4.2.11. Deductive relation $\leq_{\mathrm{bo}}$

The Boolean deduetive relation $\leq_{\mathrm{bo}} = \leq_{\mathrm{ec}\cap\mathrm{tr}}$. So

$$H_{\mathrm{bo}}(1) = H_{\mathrm{ec}\cap\mathrm{tr}}(1) = H_{\mathrm{ec}}(1) \cap H_{\mathrm{tr}}(1) = \{0, 1\} \cap \{1\} = \{1\},$$

as expected.

$$H_{\mathrm{bo}}(0) = H_{\mathrm{ec}\cap\mathrm{tr}}(0) = H_{\mathrm{ec}}(0) \cap H_{\mathrm{tr}}(0) = \{0, 1\} \cap \{0, 1, \mathrm{U}\} = \{0, 1\},$$

also as expeeted. $H_{\mathrm{bo}}(\mathrm{U}) = \{\mathrm{U}\}$ since only U has 0 condition. Finally,

$$H_{\mathrm{bo}}\{1, \mathrm{U}\} = H_{\mathrm{ec}}\{1, \mathrm{U}\} \cap H_{\mathrm{tr}}\{1, \mathrm{U}\} = \{0, 1, \mathrm{U}\} \cap \{0, 1, \mathrm{U}\} = \{0, 1, \mathrm{U}\} = \mathscr{B}/\mathscr{B}.$$

4.2.12. Deductive relation $\leq_{\mathrm{I}}$

Only equal eonditionals satisfy $\leq_{\mathrm{I}}$ and so the DCSs with respect to $\leq_{\mathrm{I}}$ are just the equivalence classes of equal conditionals, namely {0}, {1} and {U} each generated by its single conditional. The subsets {0,1}, {0,U} and {1,U} are also DCSs since the conjunction of any two of the three conditionals 0, 1, or U gives back one of the two components. So these subsets are closed under conjunction and they have only themselves as deductions with respect to $\leq_1$. The whole space {0,1,U} is also a DCS requiring all three of its conditionals to generate it.

4.2.13. Deductive relations $\leq_0$

All conditionals are deducible from any one conditional with respect to $\leq_0$. So there is just one DCS with respect to $\leq_0$, namely the whole space {0, 1, U} and it is generated by any of its members. This is true no matter what the original Boolean algebra $\mathscr{B}$. This completes the deductively closed sets of conditionals of $\mathscr{B}/\mathscr{B} = \{0, 1, \mathrm{U}\}$ with respect to the 13 deductive relations identified in the hierarchy. They are listed in the tables below. G's and H's are generators of that column, one of each present is required; + indicates inclusion in the DCS of that column; J's are joint generators, all required.

It is not so remarkable that all 13 of these preorders yield different collections of DDSs for this simplest case since even the 3-element Conditional Logic $\mathscr{B}/\mathscr{B} = \{1, 0, \mathrm{U}\}$ has eight subsets. Each deductive relation determines which of these 8 subsets will form a deductively closed set with respect to it. So there are potentially $2^8 = 256$ different possible choices of a subset of the 8 to be a DCS. These are subsets of subsets of the three original conditionals – the so-called second order predicates.

	$\leq_{ap}$		$\leq_{tr}$		$\leq_{nf}$		$\leq_{ip}$		$\leq_{ec}$			$\leq_{\vee}$			$\leq_{pm}$			$\leq_{\wedge}$		
1	G	+	+	G	G	+	G		G		G	G	+	+	G	+	+	G		+
0	G	+	G			G	G		G		G		G	+		G				G
U		G	G		G	+	+	G		G	H			G		+	G	+	G	+

	$\leq_{m\vee}$				$\leq_{m\wedge}$				$\leq_{bo}$				$\leq_1$							$\leq_0$
1	G	+	+	+	G	+		J	G	+		J	G			J	J		J	G
0		G		J		G				G		+		G		J		J	J	G
U			G	J		+	G	J			G	J			G		J	J	J	G

4.3. Generating with a Boolean algebra $\mathscr{B}$ with two atoms

A Boolean algebra $\mathscr{B}$ with two atoms is of the form $\mathscr{B} = \{0, \mathrm{b}, \mathrm{b}', 1\}$ generated by a single non-trivial event b. There are nine conditionals generated by $\mathscr{B}$:

$$\begin{aligned}\mathscr{B}/\mathscr{B} &= \{(0|1),(b|1),(b'|1),(1|1),(0|b),(1|b),(0|b'),(1|b'),(0|0)\}\\ &= \{0,b,b',1,(0|b),(1|b),(0|b'),(1|b'),U\}.\end{aligned}$$

There are $2^9 = 512$ different subsets of $\mathscr{B}/\mathscr{B}$, and a particular deductive relation will make some collection of these subsets a DCS with respect to it. There are therefore 2^{512} different collections of subsets of conditionals that are candidates for being the set of all DCSs generated by a given deductive relation, and that is just for the 4-Element Boolean algebra. Information is indeed complex!

4.3.1. *The four elementary deductive relations* $\leq_{ap}$, $\leq_{tr}$, $\leq_{nf}$, *and* $\leq_{ip}$

The DCSs in Table 1 are determined by methods similar to those used to determine the DCSs of the 2-element Boolean algebra. For example, for the preorder $\leq_{nf}$ the conditional $(b|1)$, which is b, generates the DCS $H_{nf}(b) = \{(x|y) : b \vee 1' \leq x \vee y'\} = \{(x|y) : b \leq x \vee y'\}$. If $y = 0$, then $b \leq x \vee y'$ is satisfied. So $(0|0)$ is in $H_{nf}(b)$. If $y = b'$, then the inequality is also satisfied for any x. So $(0|b')$ and $(1|b')$ are also in $H_{nf}(b)$. If $y = b$ then the inequality is satisfied if and only if $b \leq x$. So $(1|b)$ is in $H_{nf}(b)$. Finally, if $y = 1$, then again $b \leq x$, and so $(1|1)$ is in $H_{nf}(b)$. So in Table 1 all of the conditionals of DCS #3 under the deductive relation $\leq_{nf}$ have been shown to be in $H_{nf}(b)$, the DCS with respect to $\leq_{nf}$ generated by b. Now $H_{nf}(0|b') = H_{nf}(b)$ since the same inequality shows up. That is, $H_{nf}(0|b') = \{(x|y) : 0 \vee (b')' \leq x \vee y'\} = \{(x|y) : b \leq x \vee y'\}$. So $(0|b')$ generates the same DCS. Similar examination of the other conditionals generated by b with respect to $\leq_{nf}$ yields no new conditionals. For instance, $(1|b)$ generates all unity events $(1|y)$ for any y, but they are already included. In fact, $(1|b)$ generates DCS #2, which is a subsystem of DCS #3 of preorder $\leq_{nf}$.

Having determined the DCSs for the elementary preorders, the DCSs for the combination preorders can be determined with the help of the Conjunction Theorem for Deductively Closed Sets with respect to two preorders. For example, using the table, one DCS with respect to $\leq_{pm}$, which is $\leq_{tr\cap nf}$, is determined by intersecting the conditionals in DCS #3 of $\leq_{tr}$ with those in DCS #4 of $\leq_{nf}$. The result is $\{1,(1|b)\}$, the DCS with respect to $\leq_{pm}$ generated by $(1|b)$.

In fact, since the whole space $\mathscr{B}/\mathscr{B}$ is a DCS with respect to any combined preorder, a DCS with respect to one of its component preorders will also be a DCS with respect to the combined preorder by intersection of the DCS with the whole space. This common DCS may in general have different generators with respect to the two preorders, and so they have been included in the tables below.

However, some DCSs, like #11 below of preorder $\leq_{\vee}$ are not the intersection of DCSs of more elementary preorders. They were determined with the

Table 2
Deductively closed sets of $\mathscr{B}/\mathscr{B} = \{0, b, b', 1, (0|b), (1|b), (0|b'), (1|b'), U\}$ with respect to the deductive relation $\leq_{ec}$

DCS	Nine conditionals of $\mathscr{B} = \{0, b, b', 1\}$									
	#	1	b	b′	(1|b)	(1|b′)	0	(0|b)	(0|b′)	U
Condition: 1	1	G	G	G			G			
Cond: b	2				H			H		
b′	3					K			K	
0	4									L
Conds: 1,b	5	G	G	G	H		G	H		
1,b′	6	G	G	G		K	G		K	
1,0	7	G	G	G			G			L
b,b′,1	8	+	+	+	H	K	+	H	K	
b,0	9				H			H		L
b′,0	10					K			K	L
b,b′,0,1	11	+	+	+	H	K	+	H	K	L

Key. # – Numbered deductively closed sets (DCSs) of the Deductive relation; + – Included in the DCS of that row; G, H, K, L – Generators of the DCS of that row; one of each present in a row is required to generate that row.

Table 3
Deductively closed sets of $\mathscr{B}/\mathscr{B} = \{0, b, b', 1, (0|b), (1|b), (0|b'), (1|b'), U\}$ with respect to the deductive relation $\leq_{\vee}$

DCS	Nine conditionals of $\mathscr{B} = \{0, b, b', 1\}$									
	#	1	b	b′	(1|b)	(1|b′)	0	(0|b)	(0|b′)	U
	1	G								
	2	+	G							
	3	+		G						
	4	+	+		G					
	5	+		+		G				
	6	+	+	+			G			
	7	+	+	+	+		+	G		
	8	+	+	+		+	+		G	
	9	+	+	+	+	+	+	+	+	G
	10	+	+	+	+	+	+	J	J	
	11	+	+	+	J	J	+			
	12	+	H	+		G	H			
	13	+	+	H	G		H			
	14	+	+	+	J	+	+		J	
	15	+	+	+	+	J	+	J		

Key. # – Numbered deductively closed sets (DCSs) for a preorder; + – Conditional included in the DCS of that numbered row; G – Generator of the DCS of that row; any G; J – Joint generators; all conditionals with J required.

Table 4
Deductively closed sets of $\mathscr{B}/\mathscr{B} = \{0, b, b', 1, (0|b), (1|b), (0|b'), (1|b'), U\}$ with respect to the preorder $\leq_{pm}$

DCS	Nine conditionals of $\mathscr{B} = \{0, b, b', 1\}$									
	#	1	b	b′	(1|b)	(1|b′)	0	(0|b)	(0|b′)	U
	1	G								
	2	+	G		+					
	3	+		G		+				
	4	+			G					
	5	+				G				
	6	+	+	+	+	+	G	+	+	+
	7	+		+	+	+		G		+
	8	+	+		+	+			G	+
	9	+			+	+				G
	10	+			J	J				
	11	+	J		+	J				
	12	+		J	J	+				
	13	+	J		+	+				J
	14	+		J	+	+				J

Key. # – Numbered deductively closed sets (DCSs) for a preorder; + – Conditional included in the DCS of that numbered row; G – Generator of the DCS of that row; any G; J – Joint generators; all conditionals with J required.

Table 5
Deductively closed sets of $\mathscr{B}/\mathscr{B} = \{0, b, b', 1, (0|b), (1|b), (0|b'), (1|b'), U\}$ with respect to the deductive relation $\leq_{\wedge}$

DCS	Nine conditionals of $\mathscr{B} = \{0, b, b', 1\}$									
	#	1	b	b′	(1|b)	(1|b′)	0	(0|b)	(0|b′)	U
	1	G			+	+				+
	2	+	G		+	+			+	+
	3	+		G	+	+		+		+
	4				G					+
	5					G				+
	6	+	+	+	+	+	G	+	+	+
	7				+			G		+
	8					+			G	+
	9									G

Key. # – Numbered deductively closed sets (DCSs) for a preorder; + – Conditional included in the DCS of that numbered row; G – Generator of the DCS of that row.

help of the Theorem on Additional Deductive Information and previously determined principal DCSs.

4.3.2. The compound deductive relations (preorders)

Using the methods just illustrated the DCSs of the other preorders in the hierarchy were determined and are listed in the following tables:

(a) Deductive relation $\leq_{ec}$ (see Table 2).
(b) Deductive relation $\leq_{\vee}$ (see Table 3).
(c) Deductive relation $\leq_{pm}$ (see Table 4).
(d) Deductive relation $\leq_{\wedge}$ (see Table 5).
(e) Deductive relation $\leq_{m\vee}$ (see Table 6).
(f) Deductive relation $\leq_{m\wedge}$ (See Table 7).

Table 6
Deductively closed sets of $\mathscr{B}/\mathscr{B} = \{0, b, b', 1, (0|b), (1|b), (0|b'), (1|b'), U\}$ with respect to the preorder $\leq_{m\vee}$

DCS	Nine conditionals of $\mathscr{B} = \{0, b, b', 1\}$									
	#	1	b	b'	(1|b)	(1|b')	0	(0|b)	(0|b')	U
	1	G								
	2	+	G							
	3	+		G						
	4	+			G					
	5	+				G				
	6	+	+	+			G			
	7	+		+	+			G		
	8	+	+			+			G	
	9	+			+	+				G
	10	+	+	+	+		J	J		
	11	+	J		J					
	12	+	+	+		+	J		J	
	13	+		J		J				
	14	+	+	+	+	+	+	J	J	J
	15	+	+		+	+			J	J
	16	+		+	+	+		J		J
	17	+	+	+	+	+	+	J	J	
	18	+			J	J				
	19	+	+		J	+			J	
	20	+		+	+	J		J		
	21	+	+	+	J	J	J			
	22	+	J		J	J				
	23	+		J	J	J				
	24	+	+	+		J	J			
	25	+	+	+	J		J			
	26	+	J			J				
	27	+		J	J					
	28	+	J		+	+				J
	29	+		J	+	+				J
	30	+	+	+	+	+	J			J
	31	+	+	+	+	J	J	J		
	32	+	+	+	J	+	J		J	

Key. # – Numbered deductively closed sets (DCSs) for a preorder; + – Conditional included in the DCS of that numbered row; G – Generator of the DCS of that row; J – Joint generators; all conditionals with J required.

Table 7
Deductively closed sets of $\mathscr{B}/\mathscr{B} = \{0, b, b', 1, (0|b), (1|b), (0|b'), (1|b'), U\}$ with respect to the deductive relation $\leq_{m\wedge}$

DCS	Nine conditionals of $\mathscr{B} = \{0, b, b', 1\}$									
#	1	b	b′	(1|b)	(1|b′)	0	(0|b)	(0|b′)	U	
1	G									
2	+	G		+						
3	+		G		+					
4				G						
5					G					
6	+	+	+	+	+	G	+	+	+	
7				+			G		+	
8					+			G	+	
9									G	
10	J			J						
11	J				J					
12	J				+			J	+	
13	J			+			J		+	
14	+	J		+	J					
15	+		J	J	+					
16	+	G		G	+			H	+	
17	+		G	+	G		H		+	
18	+			J	J					
19	+			J	J				J	
20	+	J		+	J				J	
21	+		J	J	+				J	
22				J					J	
23					J				J	
24	J								J	
25	+	J		+					J	
26	+		J						J	

Key. # – Numbered deductively closed sets (DCSs) for a preorder; + – Conditional included in the DCS of that numbered row; G, H – Generators of the DCS of that row; one of each present in a row is required to generate that row; J – Joint generators; all conditionals with J required.

5. Summary

Boolean Deduction is simply inclusion of one uncertain event by another, but deduction between conditional events, pairs of events, is driven by the four chosen operations of "not" (negation), "or" (addition), "and" (multiplication) and "given" (division), which were extensively analyzed and justified in Sections 2.2 and 2.3. Using these operations to define deduction in the ways that are equivalent for Boolean deduction results in non-equivalent implication relations in the realm of conditionals. These implications form a hierarchy of eleven built upon some subset of four elementary ones. The deductively closed sets (DCS) of conditionals generated by the different deductive relations have been determined. For a single conditional (a|b), the principal DCS generated is just the set of all conditionals implied by (a|b) with respect to the deductive relation. Matters are much more complicated when there are two or more generators. For three of the deductive relations, the DCS generated by a finite set J of conditionals is principal and generated by the conjunction of all members of J. Except for two of the other deductive relations, the DCS generated by J is the union of the principal DCSs of the set $D(J)$ of all finite conjunctions of members of J. But for the remaining two deductive relations, this union is not necessarily a DCS. The principal DCSs are explicitly solved for all deductive relations. The results are applied to the famous penguin problem: "Birds fly", "Penguins are birds", and "Penguins don't fly" to determine the DCSs of this set of three conditionals with respect to the various types of implications. A final section provides the complete collection of DCSs with respect to all deductive relations for the conditional event algebra generated by the two-element Boolean algebra {0,1} and also by the 4-element Boolean algebra $\{0, b, b', 1\}$. These DCSs provide some concrete finite examples and counter-examples by which to view the sometimes surprising results presented above concerning deduction with uncertain conditionals.

References

[1] S. Abramsky, Domain theory in logical form, Annals of Pure and Applied Logic (1988).

[2] E.W. Adams, On the logic of conditionals, Inquiry 8 (1965) 166–197.

[3] E.W. Adams, Probability and the logic of conditionals, in: J. Hintikka, P. Suppes (Eds.), Aspects of Inductive Logic, North-Holland, Amsterdam, 1966, pp. 265–316.

[4] E.W. Adams, The Logic of Conditionals: An Application of Probability to Deductive Logic, Reidel, Boston, MA, 1975.

[5] G. Boole, An Investigation of the Laws of Thought on which are Founded the Mathematical Theories of Logic and Probabilities, second ed., Macmillan, New York, 1854 (Open Court; 1st ed.).

[6] P. Buneman, S. Davidson, A. Watters, A Semantics for Complex Objects and Approximate Queries, Department of Computer and Information Science, School of Engineering and Applied Science, University of Pennsylvania, Philadelphia, PA 19104, Report MS-CIS-87-99, 1990.

[7] P. Buneman, A. Jung, A. Ohori, Using Powerdomains to generalize relational databases, Department of Computer and Information Science, School of Engineering and Applied Science, University of Pennsylvania, Philadelphia, PA 19104, Logic and Computation 22, Report MS-CIS-90-62, 1990.

[8] P.G. Calabrese, The probability that p implies q (preliminary report), Notices American Mathematical Society 22 (3) (1975) A430–A431.

[9] P.G. Calabrese, An algebraic synthesis of the foundations of logic and probability, Information Sciences 42 (1987) 187–237.

[10] P.G. Calabrese, Reasoning with uncertainty using conditional logic and probability, in: Proceedings of the First International Symposium on Uncertainty Modeling and Analysis, IEEE Computer Society, Silver Spring, MD, 1990, pp. 682–688.

[11] P.G. Calabrese, Deduction and inference using conditional logic and probability, in: I.R. Goodman et al. (Eds.), Chapter 2 in Conditional Logic in Expert Systems, North-Holland, Amsterdam, 1991, pp. 71–100.
[12] P.G. Calabrese, An extension of the fundamental theorem of Boolean algebra to conditional propositions, Part I (1–18) of Conditional event algebras and conditional probability logics by P.G. Calabrese and I.R. Goodman, Probabilistic Methods in Expert Systems, Romano Scozzafava ed., Proceedings of the International Workshop, Rome, Italy, Societá Italiana di Statistica, 1993, pp. 1–35.
[13] P.G. Calabrese, A theory of conditional information with applications, IEEE Transactions of Systems, Man and Cybernetics 24 (Number 12) (1994) 1676–1684.
[14] P.G. Calabrese, Conditional Events: Doing for Logic and Probability What Fractions Do for Integer Arithmetic, Invited Talk, International Conference on The Notion of Event in Probabilistic Epistemology, May 27–29, 1996, Università Degli Studi di Trieste, Departimento di Matematica Applicata, Alle Scienze Economiche e Statistiche ed Attuariali, Trieste, Publicazione no. 4, September 1997, pp. 175–212.
[15] P. Chrzastowski-Wachtel, A. Hoffmann, A. Ramer, J. Tyszkiewicz, Mutual definability of connectives in conditional event algebras of Schay–Adams–Calabrese and Goodman–Nguyen–Walker, Information Processing Letters 79 (2001) 155–160.
[16] B. De Finetti, La prévision, ses lois logiques et ses sources subjectives, Ann. Inst. H. Poincaré 7, 1937 (Transl. by H. Kyburg Jr.), in: H. Kyburg Jr., H. Smokler (Eds.), Studies in Subjective Probability, Wiley, New York, 1964, pp. 95–158.
[17] B. De Finetti, Probability, Induction and Statistics, Wiley, New York, 1972.
[18] D. Dubois, H. Prade, Conditional objects as nonmonotonic consequence relationships, IEEE Transactions of Systems, Man and Cybernetics 24 (12) (1994) 1724–1740.
[19] A. Gilio, R. Scozzafava, Conditional events in probability assessment and revision, IEEE Transactions of Systems, Man and Cybernetics 24 (12) (1994) 1741–1746.
[20] I.R. Goodman, H.T. Nguyen, E.A. Walker, Conditional Inference and Logic for Intelligent Systems: A Theory of Measure-Free Conditioning, North-Holland, Amsterdam, 1991.
[21] I.R. Goodman, M.M. Gupta, H.T. Nguyen, G.S. Rogers (Eds.), Conditional Logic in Expert Systems, North-Holland, Amsterdam, 1991.
[22] C.A. Gunter, The Mixed Powerdomain, Department of Computer and Information Science, School of Engineering and Applied Science, University of Pennsylvania, Philadelphia, PA 19104, Logic and Computation 25, Report MS-CIS-90-75, Revised (Feb. 1991).
[23] T. Hailperin, Sentential Probability Logic, Lehigh University Press, 1996.
[24] R. Heckmann, Power Domains and Second Order Predicates, Study Note S.1.6-SN-25.0, Universitat des Saarlandes, PROSPECTRA Project, 6000 Saarbrucken, FRG (Feb. 1990).
[25] E. Jaynes, Probability Theory: The Logic of Science, (1994 fragmentary edition) http://omega.albany.edu:8008/JaynesBook.
[26] D. Lewis, Probabilities of conditionals and conditional probabilities, The Philosophical Review 85 (3) (1976) 297–315.
[27] G.D. Plokin, A powerdomain construction, SIAM Journal of Computing (5) (1976) 452–487.
[28] N. Rescher, Many-valued Logics, McGraw-Hill, New York, 1969, p. 342.
[29] W. Rödder, Conditional logic and the principle of entropy, Artificial Intelligence 117 (2000) 83–106.
[30] W. Rödder, C.-H. Meyer, Coherent Knowledge Processing at Maximum Entropy by SPIRIT, in: Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence, Portland, Oregon, USA, 1996, pp. 470–476.
[31] B. Russell, A.N. Whitehead, Principia Mathematica, Cambridge University Press, Cambridge, 1913.
[32] G. Schay, An algebra of conditional events, Journal of Mathematical Analysis and Applications 24 (1968) 334–344.

[33] J.E. Shore, Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy, IEEE Transactions on Information Theory, IT-26 1 (1980).
[34] M. Smyth, Power domains, Journal of Computer Sciences 16 (1978) 23–36.
[35] B. Sobocinski, Axiomatization of a partial system of the three-valued calculus of propositions, Journal of Computing Systems 1 (1952) 23–55.
[36] E.A. Walker, Stone algebras conditional events and three valued logic, IEEE Transactions of Systems, Man and Cybernetics 24 (12) (1994) 1699–1707.
[37] G. Winskel, On powerdomains and modality, Theoretical Computer Science 36 (1985) 127–137.